

RISTEYSLIIKENTEEEN SIMULOINTIMALLI KONFLIKTI-
JA ONNETTOMUUSTUTKIMUSTA VARTEN

Diplomityö
Helsingin teknillinen korkeakoulu
Teknillisen fysiikan osasto
Markku Karhu

D Ø 7

TEKNILLINEN KORKEAKOULU
TEKNILLISEN FYSIIKAN LAITOS
KIRJASTO

Aihe hyväksytty 1974-11-04

Työ jätetty tarkastettavaksi 1975-02-28

Työ tehty prof. Lokin johdolla ja

TkL Kallbergin ohjauksella

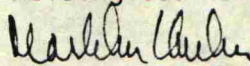
ESIPUHE

Aloitteen tähän tutkimukseen teki Valtion teknillisen tutkimuskeskuksen pääjohtaja Pekka Jauho, joka esitti harkittavaksi mm. ydinfysiikan tutkimuksissa käytetyn Monte Carlo -simulointimenetelmän soveltamista liikenne-onnettomuuksien tutkimiseen.

VTT:n tie- ja liikennelaboratoriossa onkin aloitteen johdosta valmisteltu moniosaista risteyskonflikteja ja -onnettomuuksia käsittelevää tutkimusta, jonka yhtenä osana on tässä työssä tehty risteyskonfliktien ja -onnettomuuksien simulointimallin konstruointi. Tämä työ on tehty VTT:n atk-palvelutoimistossa tie- ja liikennelaboratorion sekä atk-palvelutoimiston välisenä yhteistoimintaprojektina.

Työn ohjaajalle TkL Harri Kallbergille ja projektin vetäjälle DI Matti Huhtalalle esitän parhaimmat kiitokseni työn kuluessa saamastani opastuksesta ja arvokkaista neuvoista. Työn johtajaa prof. Olli Lökkia kiitän hänen työtäni kohtaan osoittamastaan mielenkiinnosta. VTT:n atk-palvelutoimiston henkilökuntaa ja erikoisesti esimiestäni DI Jukka Kiveä kiitän mahdollisuudesta tehdä työni atk-palvelutoimistossa.

Helsingissä 1975-02-19


Markku Karhu

SISÄLLYSLUETTELO

1	JOHDANTO	1
2	YLEISTÄ SIMULOINNISTA JA LIIKENNESIMULOINNISTA	6
2.1	Koetutkimisesta	6
2.1.1	Systeemin tutkimusmenetelmistä	6
2.1.2	Liikennesysteemin tutkimisesta	9
2.1.3	Liikennesysteemin malleista	10
2.2	Simuloinnista	12
2.2.1	Simulointitutkimuksen työjärjestys	12
2.2.2	Simulointiohjelman organisointi	14
2.2.3	Simuloinnin ajansiirtomekanismi	15
2.2.4	Simuloinnin kustannuksista	18
3	SIMULOINTIMALLIN KONSTRUOINTI	21
3.1	Simulointimallin rakenne ja sisäinen kirjanpito	21
3.2	Simuloinnin aika-askelen valinta	23
3.3	Katuristeyksen rakenteen esittäminen	24
3.3.1	Määritelmiä	24
3.3.2	Koordinaatisto	26
3.3.3	Risteyksen käsittelytekniikka	26
3.4	Ajoneuvojen esittäminen	28
3.4.1	Ajoneuvoihin ja ajajiin liittyvät ominaisuudet	28
3.4.2	Ajoneuvojen identifiointi	31
3.5	Ajoneuvojen etenemismallit	33
3.5.1	Vapaa eteneminen	35
3.5.2	Edeltäjän seuraaminen	37
3.5.3	Pysähtyminen	43
3.5.4	Kääntyminen	45
3.5.5	Jonon purkaantuminen	47
3.6	Ajajan käyttäytymismallit	49
3.6.1	Yleistä havaitsemisesta, päätöksenteosta ja reaktioajasta	49
3.6.2	Keltaista liikennevaloa lähestyminen	53
3.6.3	Päätöksenteko toista ajouraa ylitettäessä ja konfliktitilanteissa	55
3.7	Jalankulkijat ja ympäristötekijät	62
3.8	Ajoneuvojen generointi ja poistaminen risteys- systeemistä	63
3.9	Liikennesuureiden mittauksista mallilla	64

4	SIMULOINTIMALLIN OHJELMOINTI	66
4.1	Simulointikieli	66
4.2	Simulointilogiikan pääpiirteet	67
4.3	Ohjelman osien tehtävät ja itsenäisyys	68
4.3.1	Pääohjelma	68
4.3.2	Aliohjelma LUKU	69
4.3.3	Aliohjelma INKONF	70
4.3.4	Aliohjelma PAIVIT	71
4.3.5	Aliohjelma POISTU	74
4.3.6	Aliohjelma URVAIH	74
4.3.7	Aliohjelma GENERO	75
4.3.8	Aliohjelma LVALOT	76
4.3.9	Aliohjelma JONOT	76
4.3.10	Aliohjelma KIIHTY	77
4.3.11	Aliohjelma KONFLI	81
4.3.12	Aliohjelma KIRJA	84
4.3.13	Aliohjelma SATLUK	85
4.4	Koneriippuvat osat	87
4.5	Ohjelman muuntelumahdollisuudet	89
4.6	Muuttujaluettelot	90
4.7	I/O - toiminta	101
4.7.1	Syöttötietojen kuvaus	101
4.7.2	Tulostustietojen kuvaus	101
4.7.3	Virheilmoitukset	111
4.8	Atk-laitteisto ja käyttö	113
4.8.1	Atk-laitteisto	113
4.8.2	Käyttö UNIVAC 1108:ssa	113
5	SIMULOINTIMALLIN VALIDISOINTI JA JATKOTUTKIMUKSESTA	115
5.1	Validisointikriteerejä	115
5.2	Simulointiajojen suorittamisesta	117
5.3	Työn nykyvaihe ja jatkotutkimuksista	120
6	TIIVISTELMÄ	122
7	KIRJALLISUUSVIITTEET JA LISTAUKSET	124
7.1	Lähdekirjallisuus	124
7.1.1	Tekstissä referoitu lähdekirjallisuus	124
7.1.2	Referoimaton lähdekirjallisuus	128
7.2	Ohjelmayksiköiden kulkukaaviot	131
7.3	Ohjelmalistaukset	152
7.4	Esimerkkiajon tulostuslistaus	174

KUVALUETTELO*

kuva	sivu
1-1. Onnettomuuteen ja sen raportointiin johtava tapahtumaketju	3
2-1. Välitön ja välillinen koetutkiminen	7
2-2. Simulointitutkimuksen työjärjestys	13
2-3. Simulointimallin yleinen käsittelyjärjestys	14
2-4. Esimerkki tarkuuden menetyksestä	17
2-5. Esimerkki täsmällisyyden menetyksestä	17
3-1. Katuristeyksen rakenteen kooditus	25
3-2. Ketjutustekniikan käyttö fyysisen ajourajonon esittämisessä	27
3-3. Ajoneuvokohtaisten tietojen sijoittaminen tiivistettyyn sanaan	29
3-4. Ketjutustekniikan käyttö ajoneuvojen esittämisessä	31
3-5. Käyttäytymis- ja etenemismallien hierarkkinen rakenne	34
3-6. Kulmanopeuden havaitseminen	40
3-7. Reaktioajan vaikutus edeltäjän seuraamismallin stabiilisuuteen	42
3-8. Jonon purkaantuminen	48
3-9. Autonkuljettajan ajotoiminnan kolme vaihetta: havainto, ratkaisu ja suoritukset	49
3-10. Kuljettajan toiminnan eri vaiheet ja virhemahdollisuudet	51
3-11. Kuljettajan reaktioaika	52
3-12. Todennäköisyys pysähtyä keltaisissa liikennevaloissa	54
3-13. Vastaaan tulevan auton nopeuden arviointi	57
3-14. Vastaaan tulevan auton etäisyyden arviointi	57
3-15. Aikavälin ja matkavälin hyväksymisjakautumat	58
3-16. Havaitsemisen todennäköisyysfunktion graafinen esitys eräillä parametrien arvoilla	61
4-1. Kääntyvän ajouran käsittelyjärjestys	73
4-2. Ajourien käsittelyjärjestys	73
4-3. Ohjelmiston kokonaislogiikka	131
4-4. Pääohjelman käsittelylogiikka	132
4-5. Aliohjelman LUKU kulkukaavio	133
4-6. Aliohjelman INKONF kulkukaavio	134
4-7. Aliohjelman PAIVIT käsittelylogiikka	135

4-8. Aliohjelman POISTU käsittelylogiikka	136
4-9. Aliohjelman URVAIH käsittelylogiikka	137
4-10. Aliohjelman GENERO käsittelylogiikka	138
4-11. Aliohjelman LVALOT käsittelylogiikka	139
4-12. Aliohjelman JONOT käsittelylogiikka	140
4-13. Aliohjelman KIIHTY käsittelylogiikka	141
4-14. Alirutiinin VIHKEK käsittelylogiikka	142
4-15. Alirutiinin PUNKEL käsittelylogiikka	143
4-16. Alirutiinin PUNLAH käsittelylogiikka	144
4-17. Alirutiinin KAANTY käsittelylogiikka	145
4-18. Aliohjelman KONFLI käsittelylogiikka	146 ja 147
4-19. Alifunktion HYVVAL käsittelylogiikka	148
4-20. Alifunktion XTOD käsittelylogiikka	149
4-21. Alirutiinin JALKNF käsittelylogiikka	150
4-22. Aliohjelman KIRJA käsittelylogiikka	151
4-23. Ajovirtakaavio	114
7-1. Esimerkkiajon risteyksen kooditus	174

*Kuvat 4-3...4-22 on sijoitettu tutkimuksen liiteosaan (kohta 7.2)

TAULUKKOLUETTELO

taulukko	sivu
2-1. Eri liikennetutkimusmenetelmien vertailu	9
2-2. Vertailu suhteellisista simulointiajokustannuksista käytettäessä eri tietokonejärjestelmiä	20
3-1. Päätöstaulu ajoneuvon identifiointitaulukoihin tehtävistä muutoksista ajoneuvon siirtäessä uuteen ajouraan	32 ja 33
3-2. Ajotapahtumat ja niiden esiintymistiheydet vilkkaassa liikenteessä	50
3-3. Todennäköisyys pysähtyä keltaisissa liikennevaloissa	54
3-4. Aikavälin ja matkavälin hyväksymisen todennäköisyydestä	59
4-1. Pääohjelman ja muiden ohjelmayksiköiden välinen tiedonsiirto	91
4-2. Aliohjelman LUKU ja muiden ohjelmayksiköiden välinen tiedonsiirto	91
4-3. Aliohjelman INKONF ja muiden ohjelmayksiköiden välinen tiedonsiirto	92
4-4. Aliohjelman PAIVIT ja muiden ohjelmayksiköiden välinen tiedonsiirto	92
4-5. Aliohjelman POISTU ja muiden ohjelmayksiköiden välinen tiedonsiirto	93
4-6. Aliohjelman URVAIH ja muiden ohjelmayksiköiden välinen tiedonsiirto	93
4-7. Aliohjelman GENERO ja muiden ohjelmayksiköiden välinen tiedonsiirto	94
4-8. Aliohjelman LVALOT ja muiden ohjelmayksiköiden välinen tiedonsiirto	94
4-9. Aliohjelman JONOT ja muiden ohjelmayksiköiden välinen tiedonsiirto	95
4-10. Aliohjelman KIIHTY ja muiden ohjelmayksiköiden välinen tiedonsiirto	95
4-11. Aliohjelma KONFLI ja muiden ohjelmayksiköiden välinen tiedonsiirto	96
4-12. Aliohjelman KIRJA ja muiden ohjelmayksiköiden välinen tiedonsiirto	97
4-13. Aliohjelman SATLUK ja muiden ohjelmayksiköiden välinen tiedonsiirto	96
4-14(osa 1). Koko ohjelmiston yhteiset muuttujanimet	98
4-14(osa 2). Ohjelmayksiköiden omat muuttujanimet	99 ja 100

4-15. Syöttötiedot	103...106
4-16. Tulostustiedot	107
4-17. Tiedoston FKNF määrittely ja tietueen kuvaus	108
4-18. Tiedoston FAUTO määrittely ja tietueen kuvaus	109
4-19. Tiedoston FKUVA määrittely ja tietueen kuvaus	110
4-20. Virheilmoitukset	111 ja 112

1.

JOHDANTO

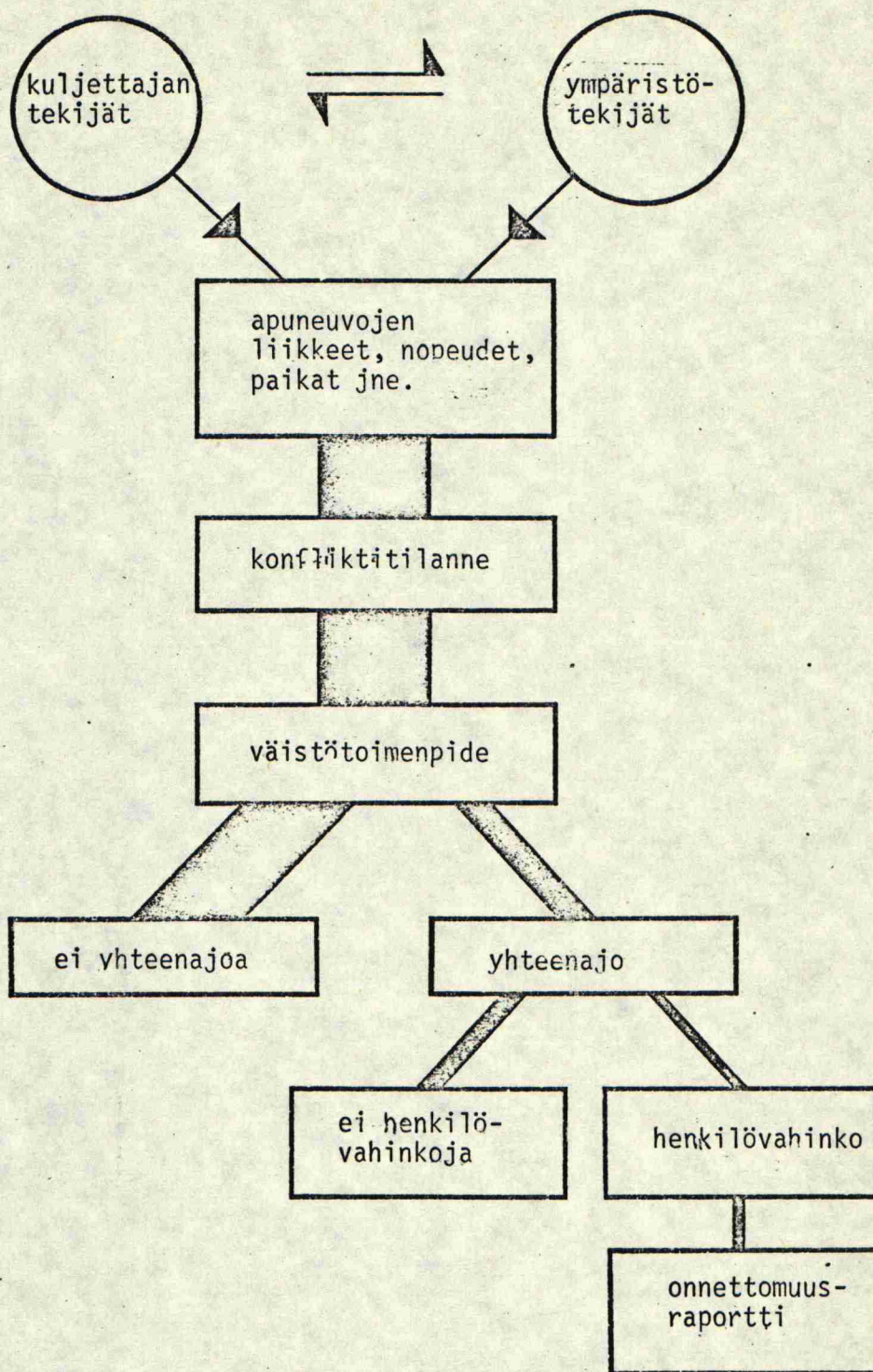
Suurin osa kaupungissa tapahtuvista liikenneonnettomuuksista sattuu katuristeyksissä. Kuitenkin katuristeyksiä suunniteltaessa ja eri vaihtoehtoja verrattaessa saadaan lasketuksi vain välityskyky ja palvelutaso sekä rakentamiskustannukset, mutta eri ratkaisujen vaikutusta liikenneturvallisuuteen voidaan vain arvioida suhteellisen hatarin perustein: mitään varsinaista vertailumenetelmää ei ole. Risteyksien liikenneturvallisuuden selvittämiseksi ei voida luonnollisestikaan tehdä kokeita ja vain onnettomuustilastoista ja erityisesti risteyksien muutostöiden yhteydessä tehtävistä ennen - jälkeen vertailuista saadaan viitteitä eri järjestelyiden vaikutuksesta liikenneturvallisuuteen. Näitäkin vaikeuttaa se, että risteystä muutettaessa samalla vaikutetaan useihin muuttujiin eikä yksityisiä tekijöitä saada selville.

Risteysonnettomuuksien tutkimista vaikeuttaa myös se, että ne ovat tutkimusmielessä harvinaisia tapahtumia, jolloin eri syiden vaikutusten analysoinnin luotettavuus tilastollisin menetelmin on vähäinen pienen havaintoaineiston vuoksi. Sen sijaan mikäli pystytään systeemiä - onnettomuus-olosuhteita ja -tapahtumia - jäljittämään mallilla eli simuloimaan, voidaan kokeilla erilaisten risteys- ja liikenneohjausjärjestelyiden vaikutusta liikenneturvallisuuteen ja samalla risteyskapasiteettiin ja odotusaikoihin. Monte Carlo-simuloinnissa konstruoidaan tietokoneeseen risteys eri toimintoineen ja laitetaan ajoneuvot liikkumaan ja käyttäytymään paitsi fyysisen lakien mukaan niin myös niille annettujen eri ominaisuuksien (esim. nopeus, liikennevalojen noudattaminen) mukaan todennäköisyysjakautumien puitteissa arpomalla.

Koska onnettomuus on tilastollisessa mielessä harvinainen tapahtuma, on pyritty tutkimaan "lähes onnettomuuksia", joiden määrittely on kuitenkin todettu hyvin subjektiiviseksi. Tästä on edelleen kehitetty käsite liikennekonflikti, jolla tarkoitetaan tapahtumia, joissa todellinen ajokäyttäytyminen poikkeaa suunnitellusta, esim. etuajo-oikeutettua tietä ajava ajoneuvo joutuu jarruttamaan tai väistämään sivulta tulevan ajoneuvon tullessa ajoradalle. Useissa tutkimuksissa, esim. /31, 33/ on saatu konfliktien ja onnettomuuksien määrien välille hyvä korrelaatio, joskin päinvastaisiakin tuloksia on olemassa. Näissä kuitenkin konfliktinkin määrittely ja onnettomuustiedot ovat olleet puutteelliset. Parempi tulos saadaan, jos kaupunkiristeysten tutkimuksissa konfliktit määritellään kussakin tapauksessa erikseen ja jaetaan luokkiin vakavuusasteen mukaan. Tässä työssä konfliktisuutta on ajateltu mitattavaksi ajajan toiminnan "hätäisyyden" avulla. Toiminnan hätäisyys on puolestaan helposti mitattavissa jarrutushidastuvuuksien avulla. Konflikteiksi voidaan määritellä "yhden tai useamman ajoneuvon väliset tilanteet, joissa on ilmeinen yhteentörmäyksen vaara, jos ajoneuvojen liike pysyisi muuttumattomana" /33/. Yksi tai useampi ajoneuvon tekemä väistöliike ratkaisee useimmat konfliktitilanteet ja vain muutamat johtavat onnettomuuteen.

Konfliktitutkimuksen mahdollisuudet onnettomuustilanteiden analysoinnissa voidaan helposti ymmärtää kuvan 1-1 avulla, jossa esitetään tapahtumasarja, joka johtaa sellaiseen onnettomuuteen, että siitä on saatavissa onnettomuusraportti. Kuvan 1-1 mukaan konfliktitilanteiden määrä on moninkertainen onnettomuuksien määrään verrattuna ja täten tilastollisen havaintoaineiston kerääminen on suoritettavissa suoraa havainnointeko- tai filmaustekniikkaa käyttäen.

Konfliktitutkimuksessa toisaalta käytetään konfliktien tapahtumisfrekvenssiä tutkittavan risteyksen turvallisuuden kriteerinä ja toisaalta konfliktiin johtavien tilanteiden yksi-



Kuva 1 - 1. Onnettomuuteen ja sen raportointiin johtava tapahtumaketju.

tyiskohtaisessa tutkimisessa käytetään pistehavaitsemista joko suoraan havaiten tai filmausta hyväksikäyttäen. Molemmat tutkimuskohteet soveltuvat hyvin simuloitaviksi, jolloin simulointimallia voitaisiin käyttää vaihtoehtoisten risteys suunnitelmien vertailuun, jossa indikaattoreina olisi liikenneturvallisuus, kapasiteetti ja odotusajat. Simulaattori toimisi eräänlaisena risteys turvallisuuden mittarina.

Kokonaisuudessaan Valtion teknillisen tutkimuskeskuksen tie- ja liikennelaboratoriossa valmisteltu useampiosainen risteyskonflikteja ja -onnettomuuksia käsittelevä tutkimus voidaan jakaa seuraaviin vaiheisiin:

1. Tietokonemallin tekeminen ja sen herkkyysanalyysi.
2. Eräiden liikennekäyttämistietojen kerääminen.
3. Muutaman katuristeyskonfliktitutkimus.
4. Onnettomuustietojen ja konfliktien keskinäisen vuorovaikutuksen selvittely.
5. Mallin antamien ja havaittujen konfliktien sekä onnettomuuksien keskinäinen vertailu.
6. Vertailun perusteella tehtävä mallin tarkistus.
7. Mallin käyttö liikennesuunnitteluun.

Tämän työn tarkoituksena on kehittää ja ohjelmoida katuristeysliikennettä simuloiva malli sekä selvittää Monte Carlo-simulointimenetelmän käyttökelpoisuutta myöhemmin suoritettavaksi suunnitellussa tutkimuksessa, joka käsittelee risteyskonfliktien ja -onnettomuuksien analysointia. Tämä työ rajoittuu siis vain koko tutkimuksen ensimmäiseen vaiheeseen.

Työn alkuvaiheessa suoritettiin liikennesimulointeja käsittelevä laajahko kirjallisuustutkimus (kohta 7.1) pyrkimyksenä selvittää, onko aikaisemmin missään suoritettu liikenneonnettomuuksien tai -konfliktien simulointia. Tulos oli kielteinen. Lukuisissa liikenneverkkotutkimuksissa on luonnollisesti simuloitu myös liittymiä ja risteyskohtia. Näissä simulointitutki-

muksissa käytettyjä eri simulaattoreiden komponentteina olevia osamalleja käytettiin hyväksi työn toisessa vaiheessa tätä simulointimallia konstruoidessa. Osamallien konstruoinnin ja validisoinnin edellytyksenä on kentällä havaitusta todellisesta liikennekäyttäytymisestä mitattujen suureiden olemassaolo. Koska tämän projektin puitteissa kenttätutkimusten suorittaminen ei ollut edes mahdollista, osamallit perustuvat kirjallisuudessa esitettyihin malleihin.

Simulaattori rakennettiin niin yleiseksi kuin mahdollista sekä risteyksen geometrisen muodon että muun syöttötiedon suhteen. Simulaattorissa käytetty risteyksen geometrisen muodon esitystapaa ei ole tietävästi käytetty missään muissa liikennesimuloinneissa, joissa risteyksen geometrinen muoto yleensä on hyvin yksinkertainen ja usein kiinteästi ohjelmoitu simulaattorin rakenteeseen.

Työn viimeisenä vaiheena suoritettiin mallilla herkkyyssanalyysi, jossa tutkittiin eräiden parametrien ja hypoteesien vaikutuksen voimakkuutta mallin käyttäytymiseen.

Projektin toteuttamiseen käytettiin n. 900 tuntia tehollista työaikaa varsinaisen mallin atk-suunnittelun ja -toteuttamisen osalta sekä lisäksi projektiin osallistuvien liikennetekniikan asiantuntijoiden työpanos lähinnä ohjaajina. Tietokoneresurssi on kulunut Valtion tietokonekeskuksen veloittaman kaupallisen hinnan mukaan laskien noin 17000 mk:n edestä.

2. YLEISTÄ SIMULOINNISTA JA LIIKENNESIMULOINNISTA

2.1 Koetutkimisesta

2.1.1 Systeemin tutkimusmenetelmistä

Käsite systeemi tarkoittaa /5/:n mukaan "joukkoa objekteja, joita yhdistää jonkinlainen säännöllinen vuorovaikutus tai -riippuvuus". Systeemissä tapahtunut muutos on objektien ominaisuuksissa tai objektien välisissä relaatioissa tapahtunut muutos. Kun tarkastellaan fysikaalista systeemiä tai hypoteettisen systeemin kuvaa, huomataan tiettyjä selviä objekteja kuten koneita, asiakkaita, sanomia jne. Kullakin objektilla on tiettyjä mielenkiinnon kohteena olevia ominaisuuksia, joita voidaan käyttää sen määrittelemiseksi: koneen suorituskky, asiakkaan tilin saldo tai sanoman pituus.

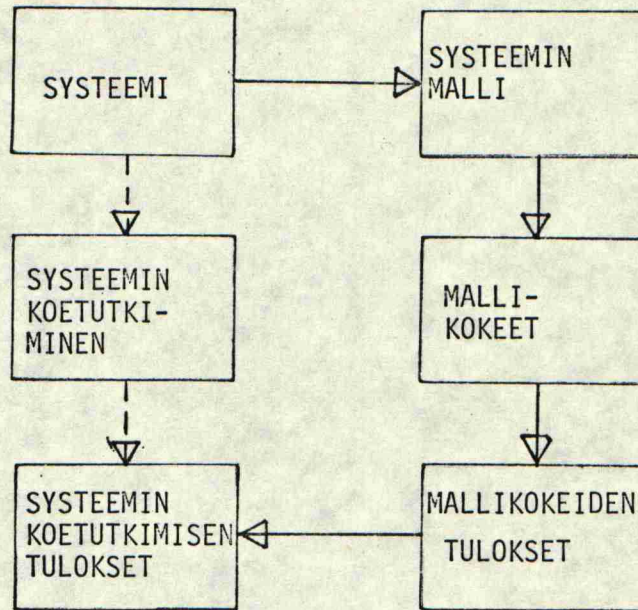
Tärkeintä systeemianalyysissä kuitenkin on havaita tietyt muutokset, jotka syntyvät systeemin tapahtumien seurauksena. Nämä muutokset esim. koneen kuormaaminen, maksun suorittaminen tai sanoman lähettäminen, kuvaavat objektien välisiä relaatioita ja määrittävät niiden ominaisuuksien arvojen muutokset.

Systeemin tutkimukselle on aina asetettavissa kaksi tavoitetta:

1. Halutaan voida ennustaa systeemin käyttäytymistä ennen kuin systeemiä on rakennettu.
2. Halutaan varmuus siitä, että toteutettava systeemin suunnitelma on optimaalinen asetettujen suunnittelukriteerien suhteen.

Systeemin tutkiminen suoritetaan yleensä käyttäen jonkinlaista systeemistä tehtyä esikuvaa, jota sanotaan malliksi. Työskentelemällä oikein konstruoidun mallin kanssa on tehtävissä tutkittavaa systeemiä koskevia päätelmiä systeimin mallilla suoritettujen kokeiden perusteella (kuva 2-1) ja täten ensimmäinen suunnittelun tavoite on toteutettu. Mallien käyttö

merkitsee systeemillä suoritettavien välittömien kokeiden muuttumisen välillisiksi kokeiksi. Sekä fysikaalisia että matemaattisia malleja voidaan käyttää hyväksi.



Kuva 2 - 1. Välitön ja välillinen koetutkiminen.

Systeemin fysikaaliset mallit esittävät tutkittavan systeemin toisella konkreettisella fysikaalisella systeemillä, joka voi olla joko pienoismalli tai analoginen malli. Fysikaalisten mallien rakentaminen on kallista ja aikaa vievää varsinkin em. toisen suunnittelutavoitteen saavuttamisessa, mikä vaatii lukuisten erilaisten mallien rakentamisen.

Matemaattisissa malleissa systeemin ominaisuudet ja objektit ovat identifioitavissa matemaattisilla muuttujilla ja objektien väliset suhteet matemaattisilla relaatioilla. Tällöin mallin ratkaisussa ja systeemin tutkimisessa voidaan käyttää analyyttisiä menetelmiä, jolloin voidaan helposti määrittää parametrien arvot, jotka vastaavat optimaalista systeemiä. Kompleksisissa systeemeissä analyyttiset menetelmät usein tulevat hyvin hankaloiksi, jos ollenkaan ovat mahdollisia.

Kun analyyttiset menetelmät eivät ole mahdollisia, systeemin toiminnasta halutun informaation saamiseksi voidaan käyttää matemaattisia malleja seuraamalla systeemiin vaikuttavien toisiaan seuraavien tapahtumien vaikutusta systeemin tilan muutokseen. Tämä voidaan toteuttaa käyttämällä numeerisia menetelmiä matemaattisen mallin vastaavien muutosten seuraamisessa. Menetelmää kutsutaan digitaaliseksi simuloinniksi.

"Toisiaan seuraavat tapahtumat" johtavat siihen, että aika on otettava mukaan yhdeksi riippumattomaksi muuttujaksi ja on rajoitettava vain tietyn tyyppisiin systeemeihin, joihin tosin kuuluu suuri joukko keskeisiä systeemejä laajalta sovellutusalueelta.

Mallirakentamisessa on mahdollista käyttää hyväksi niitä kaikkia objektien välisiä suhteita, jotka voidaan kuvata tyydyttävästi matemaattisilla tai loogisilla relaatioilla. Digitaalinen simulointi on erikoisen sopiva sellaisten systeemien tutkimiseen, joissa mallirelaatiot eivät ole deterministisiä ja joidenkin muuttujien käyttäytyminen on kuvattavissa todennäköisyysjakautumien avulla. Digitaalisessa simuloinnissa on suhteellisen helppo käsitellä stokastisia muuttujia ja aina niitä tarvittaessa suorittaa satunnaisosotus (arpominen). Jos satunnaisprosesseja käytetään, simulointi riippuu tilastollisten päätelmien ja koesuunnittelun toteutuksesta todennäköisyysjakautumien estimointia ja tulosten tulkintaa apuna käyttäen.

Mallin tarkkuuden rajoissa digitaalisessa simuloinnissa on ennustettavissa systeemin toiminta jo ennen todellisen systeemin rakentamista. Rajoitettu määrä kokeita ei koskaan takaa optimaalista järjestelmää, mutta digitaalisen simuloinnin taloudellisuuden ja nopeuden vuoksi on suoritettavissa useampia yrityksiä ja systeemin parannuksia kuin fysikaalisilla malleilla olisi mahdollista suorittaa.

Simulointi on voimakas väline, mutta kuten muutkin voimakkaat välineet, se voi olla vaarallinen väärissä käsissä. Kasvava

kiinnostus simulointimenetelmää kohtaan ja samalla vastaavan kokemuksen puute voi johtaa menetelmän käyttäjät pseudosimulointiin. Simuloinnissa on salakuoppansa niinkuin muissakin menetelmissä ihmisen yrittäessä rakentaa käsitteellistä ja idealisoitua mallia. Salakuoppien välttämiseksi (1) oletukset pitäisi tehdä vasta sitten kun niiden vaikutus on selvästi määriteltä, (2) systeemiin ei pitäisi liittää yhtään muuttujaa ilman, että kukin niistä on oikein selitetty ja sen suhde toisiin muuttujiin on määrätty ja ymmärretty ja (3) on muistettava, että yksinkertaistus on hyödyllistä, mutta liika yksinkertaistus on kohtalokasta /29/.

2.1.2 Liikennesysteemin tutkimisesta

Kuten kohdassa 2.1.1 esitettiin, systeemiä voidaan tutkia kolmella toisistaan selvästi poikkeavalla menetelmällä. Pidettäessä systeiminä tieliikennettä näiden kolmen tutkimusmenetelmän soveltuvuutta liikennetutkimuksiin on verrattu muutaman kriteerin perusteella (taulukko 2-1), joka on esitetty mm. /18/ssa). Liikennetutkimuksissa fysikaaliset mallit ovat yleensä täysimittakaavaisia kokeita tai kenttätutkimuksia.

Taulukko 2 - 1. Eri liikennetutkimusmenetelmien vertailu.

kriteeri	analyttiset menetelmät	digitaalinen simulointi	kenttäkokeet
hinta	pieni	välillä	suuri
aika	pieni	välillä	suuri
muunneltavuus	suuri	välillä	pieni
totuudenmukaisuus	pieni	välillä	suuri
tulosten yleisyys	suuri	välillä	pieni

Analyttisten menetelmien käyttöä liikennetutkimuksissa rajoittaa se, että liikennesysteemit ovat siksi monimutkaisia, että useinkin koko systeemin käsitteleminen matemaattisella mallilla osoittautuu mahdottomaksi. Jo yksinkertaisissa osasysteemeissä

usein joudutaan tekemään voimakkaita approksimaatioita, minkä vuoksi analyttisten menetelmien totuudenmukaisuus on pieni.

Kenttäkokeiden käyttökelpoisuutta puolestaan rajoittaa niiden toteuttamisen hankaluus ja kalleus ja se, että kohdan 2.1.1 mukaisen systeemin tutkimisen toinen tavoite on saavuttamattomissa.

Liikennetutkimuksen probleemoissa tutkittava systeemi tulee usein niin kompleksiseksi, että kumpikaan äärimenetelmä ei ole sovellettavissa, jolloin simulointi jää ainoaksi käyväksi menetelmäksi. Liikenneprosessin simuloinnilla saavutettavat päämäärät ovat selvät ja se on hintansa arvoinen. Simulointimenetelmä ei ole vain eräs keino toteuttaa sitä, mitä nykyisen tietämyksen mukaan voidaan tehdä, vaan sellaisten probleemien ratkaisuväline, joita tähän mennessä ei ole ratkaistu.

2.1.3

Liikennesysteemin malleista

Ajoneuvoliikennevirran analysoinnissa käytetyt teoriat ja menetelmät on luokiteltavissa neljään ryhmään:

1. analyttiset eli deterministiset mallit,
2. tilastolliset eli todennäköisyysmallit,
3. kontinuumi- eli hydrodynaamiset mallit ja
4. Monte Carlo-simulointimallit.

Ensimmäisessä mallityypissä käsitellään yksinäisten ajoneuvojen liikettä ja ajoneuvojen vuorovaikutuksellista liikettä toinen toisistaan niiden edetessä peräkkäin samaan suuntaan. Liike kuvataan ajoneuvojen geometrysten ja fysikaalisten ominaisuuksien ja kuljettajan oletetun käyttäytymisen avulla. Kohdassa 3.5.2 esitettävä edeltäjän seuraamismalli on esimerkki analyttisistä malleista.

Toisessa mallityypissä ajoneuvojen ja kuljettajien ominaisuuksien oletetaan muuttuvan ajoneuvon sijainnin mukaan. Tässä stokastisessa käsittelytavassa todennäköisyyslaskennan ja tilas-

totieteen menetelmiä käytetään hyväksi. Näissä malleissa jono-teoriaa on käytetty risteyksissä esiintyvien viiveitä ja jonoja karakterisoivien suureiden estimointiin. Yleensä tämä teoria on sopimaton kompleksisten liikennevirtojen käsittelyyn, sillä se edellyttää ajoneuvojen kulkevan samalla nopeudella ja tulevan systeemiin samasta kohtaa. Jonoteoriaa on kuitenkin sovellettu onnistuneesti silloin, kun liikenne on todella jono-muotoista, nopeus tasaista ja ajajan päätöksentekomahdollisuudet ovat pienet. Muut jonoprobleemat ovat mahdottomia analyytiselle mallille, sillä risteys sisältää aina transient-tilanteen.

Kolmannessa mallityypissä liikennevirtausta käsitellään hydrodynamiikan termein. Liikenteen oletetaan olevan kokonpuristuvan nesteen kaltaista ja sille oletetaan tietty tiheys ja virtausnopeus. Tätä teoriaa on käytetty lähinnä kuvaamaan jonon purkaantumisilmiötä (kohta 3.5.5) ja "pullonkaula"-tilanteita suorilla ajoväylillä.

Neljäs liikennevirran käsittelymallityyppi on enemmänkin tekniikka kuin teoria. Se perustuu reaali maailman liikennetilanteiden liikennevirtailmiöitä kuvaavien todennäköisyysjakautumien käyttämiseen satunnaisotoksia suorittamalla eli arpomiseen. Kolmea ensimmäistä käsittelyfilosofiaa rajoittaa se, että periaatteessa ne pystyvät kuvaamaan vain yksinkertaisen ja yksisuuntaisen liikenteen etenemistä. Monikaistaisen risteyksen toiminnan kuvaamiseen tarvitaan välttämättä laajempi mallifilosofia. Kun tällainen yleinen malli rakentuu yhden tai useamman em. teorian mukaisista osamalleista, joilla kuvataan tietyn erikoistilanteen mikro-aspekteja, mallin perusrakenne voidaan kehittää vain käyttämällä Monte Carlo -simulointimenetelmää, jolla voidaan luoda reaalisysteemin dynaaminen esitys. Tämä on toteutettavissa rakentamalla tietokonemalli, jota voidaan prosessoida dynaamisesti. Tietokonemallihan muodostuu useista matemaattisista osamalleista, jotka kuvaavat kuljettaja-ajoneuvoyhdistelmän operatiivisia ominaisuuksia risteysliikenteessä. Tietokoneen varusohjelmiston (software) nopea kehitys ja prosessoinnin nopeuden huima kasvu ovat mahdollistaneet komplek-

sisen risteysysteemin simulointimallin kehittämisen.

Risteys­simulointimallissa ajoneuvot kulkevat systeemin läpi ennalta määrättyjä päätössääntöjä noudattaen. Nämä päätössäännöt kuvaavat liikenteen todellista käyttäytymistä tietyissä odotettavissa olevissa tilanteissa. Koska tällaisesta mallista saadaan liikenteen toiminnasta mitä tietoa tahansa mistä risteys­ksen kohdasta tahansa ja minä ajanhetkenä tahansa, malli on suhteellisen helposti käytettävissä optimaalisen risteys­suunnitteluvaihtoehdon etsimisessä. Koska tietokonemallin keskeinen piirre on tutkimuksen toistaminen, operatiivisten olosuhteiden muutosten vaikutukset ovat testattavissa yksinkertaisesti muuttamalla lähtötietoparametreja, esim. liikennemäärää, liikenne­komponenttien suhteellisia osuuksia, risteys­ksen geometriaa tai kuljettajan ominaisuuksia. Vastaavat vaikutukset tutkittavaan suureeseen esim. liikennevirtaan tai konfliktisuuteen voidaan johtaa simulaattorin tulostuksen muutoksista käyttäen karakterisoivien suureiden kvantitatiivisia mittauksia: kulkuai­kaa, nopeuden ja liikennemäärän suhdetta tai hidastuvuus­jakautumia. Tällainen tietokonemalli on valmis käytettäväksi eri parametrien vaikutusten arvioinnissa, mikä olisi jopa mahdotonta suorittaa kenttätutkimuksin.

2.2 Simuloinnista

2.2.1 Simulointitutkimuksen työjärjestys

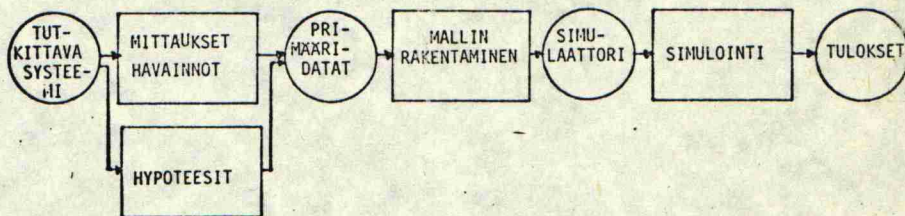
Simulointitutkimuksen työmenetelmät ovat esitettävissä kuvan 2-2 kaaviona /2/. Kaavio antaa kuvan probleeman ratkaisussa käytettävästä oikeasta työjärjestyksestä.

Toisaalta on huomattava, että simuloinnille on luonteenomaista iteratiivisuus jokaisessa sen vaiheessa, vaikka se ei eksplisiittisesti näy­kään kuvasta 2-2, johon voitaisiin piirtää lisäksi eri työvaiheisiin ja työvaiheiden välillekin takaisinkytkentää. Simulointimallin rakentaminen on

iteratiivista, koska ei ole mahdollista suoraan rakentaa sellaista mallia, joka yksityiskohdissaan sopivan tarkasti takaisi mallin oikean käyttäytymisen. Onkin tavallista alkaa yksinkertaisesta mallista ja lisätä yksityiskohtia, kunnes mallin ja todellisen systeemin välinen "samankaltaisuus" on saavuttanut "riittävän" tason.

Simulaattorin ohjelmoinnissa ohjelmien teko oikein toimiviksi vaatii omat iteratiiviset työmenetelmänsä, sillä virheettömien ohjelmien teko suoraan on mahdotonta.

Tutkittavien probleemojen ratkaisemisessa eli kohdan 2.1 toisen tavoitteen saavuttamisessa simulointimalleja käytetään iteratiivisesti. Simulointi on kokeellinen menetelmä, jossa ratkaisu löydetään vasta, kun on suoritettu useita kokeita, joiden kunkin antaman mahdollisen ratkaisun suunta ohjaa optimaalisen tai paremman ratkaisun etsimisessä.



Kuva 2 - 2. Simulointitutkimuksen työjärjestys.

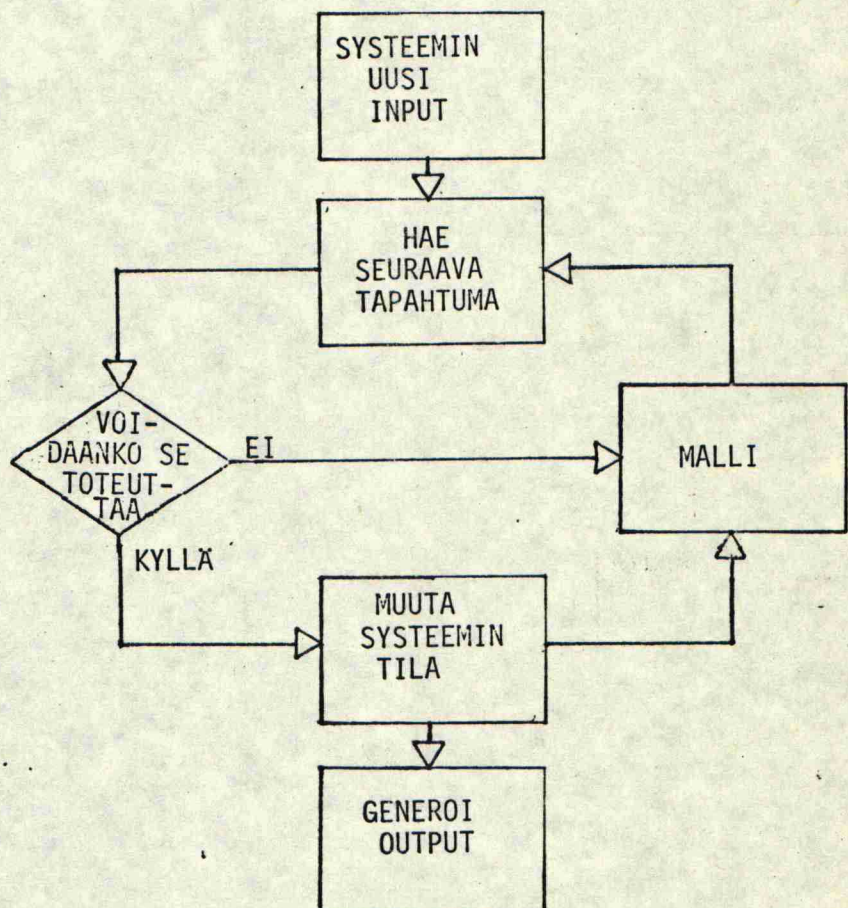
Tämän työn suorittamisessa kuvan 2-2 mukaista työjärjestystä ei valitettavasti ole voitu noudattaa, tai tarkemmin tämän työn toimeksianto sisälsi vain mallin rakentamisen ja simulaattorin ohjelmoinnin kirjallisuudesta haettavien primääridatojen ja hypoteesien pohjalta. Simulaattorin kalibrointi todelliseen ympäristöönsä eli mallin parametrien

valitseminen mittauksien ja havaintojen teon jälkeen sekä varsinainen simulointi jäävät siten tämän työn ulkopuolelle. Kuva 2-2 osoittaa, että kenttätutkimukset eli fysikaalisen mallin hyväksikäyttö on välttämätöntä simuloinnin suorittamiselle. Samoin mallin rakentamisessa matemaattisten mallien mukaan otto on yhtä välttämätöntä. Itse asiassa simuloinnin voima perustuukin kahden ääritutkimusmenetelmän yhdistämiseen.

2.2.2

Simulointiohjelman organisointi

Simulointiohjelma on organisoitavissa usealla tavalla, mutta simulointiohjelman suorittamat toiminnot yleisesti voidaan esittää kuvan 2-3 avulla. Systeemin mallia käsi-



Kuva 2 - 3. Simulointimallin yleinen käsittelyjärjestys.

tellessään ohjelma ensin etsii seuraavan systeemissä esiintyvän potentiaalisen tapahtuman, jonka jälkeen ohjelman on tutkittava systeemissä vaikuttavat relaatiot ja määritettävä, onko tapahtuma suoritettava. Jos on suoritettava, niin mallin tila muuttuu. Jos ei ole suoritettava, ohjelma etenee seuraavaan potentiaaliseen tapahtumaan ja suorittamatta jäänyt tapahtuma saattaa tulla suoritetuksi myöhemmin, kun systeemin olosuhteet ovat muuttuneet.

Liikenne-risteyksen simulointiohjelma on ajateltavissa järjestetyksi siten, että kukin looginen toimintakokonaisuus muodostaa aina oman kierrosmallinsa kuvan 2-3 mukaan. Kukin looginen yksikkö käyttää uutena syötteenään edellisen toimintayksikön muodostamaa tulostetta. Mallissa esiintyvien tapahtumien suorittamismahdollisuutta tutkitaan yksitellen ja tapahtuman tapahduttua mallin tilaa muutetaan vastaavasti, jolloin samalla ajanhetkellä tutkittavan seuraavan tapahtuman suoritushmahdollisuus riippuu mallin uudesta tilasta.

Mallin tilan muuttuessa voi olla tarpeellista päivittää tilastoja, jotka ovat osa systeemin toimintaa kuvaavaa tulostusta. Kuvan 2-3 kierrosta toteutetaan niin kauan kuin on simuloinnin suorittamiselle tarpeellista.

2.2.3

Simuloinnin ajansiirtomekanismi

Ajan kulun esittämiseen simuloinnissa on olemassa kaksi perusmenetelmää /24, 25/, jotka vaikuttavat paljon systeemin käsittelyn organisointiin. Toisissa simuloinneissa kelloa siirretään tasaisin aikavälein (ns. aika-askelmenetelmä). Systeemiä käsiteltäessä sitten etsitään ne tapahtumat, jotka uuden kelloajan mukaan on määrä tapahtua. Toisessa menetelmässä (ns. tapahtuma-askelmenetelmässä)

kelloa siirretään seuraavaksi esiintyvän merkittävän tapahtuman tapahtumishetkeen. Tavallisesti tarvitaan yksi systeemin käsittely seuraavan kelloajan identifioimiseksi ja toinen käsittely niiden tapahtumien etsimiseksi, jotka on määrä sattua uudella "kellonlyönnillä". Sellaisissa systeemeissä, joissa tapahtumien oletetaan esiintyvän säännöllisesti, ensimmäinen menetelmä on tavallisesti tehokkaampi. Jos taas tapahtumat esiintyvät ajan suhteen epäsäännöllisesti, toinen menetelmä on tavallisesti tehokkaampi.

Suoritettaessa tarkempaa kahden ajansiirtomekanismin välistä vertailua, voidaan vertailukriteereiksi ottaa

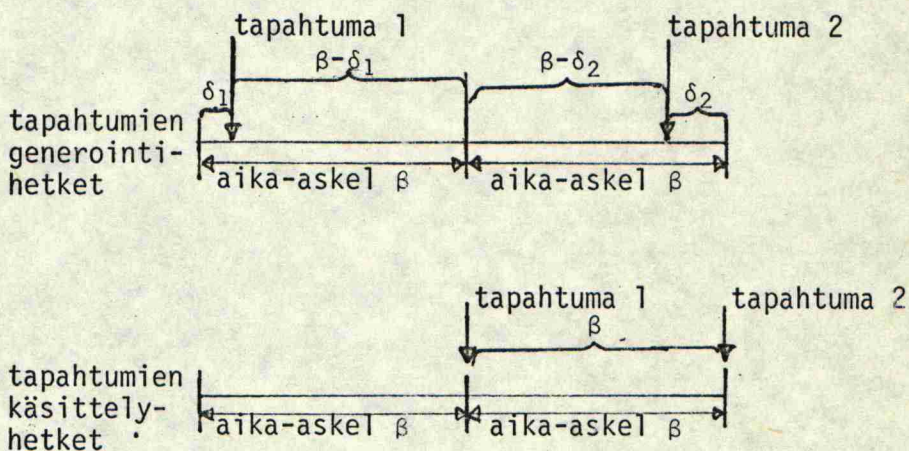
1. Simuloinnin tulosten täsmällisyys ja tarkkuus ohjelman käsittelyrutiinien tehokkuuden suhteen.
2. Tietokoneajoaika.
3. Ohjelman tarvitsemien muistipaikkojen tarve.
4. Ohjelmointityön tarve.

Menetelmien perusero on se, että aika-askel-menetelmässä systeemi käsitellään ennaltamäärätyin aikaväleihin, jotka eivät välttämättä ole, mutta tavallisesti ovat yhtäpitkiä, kun taas tapahtuma-askel-menetelmässä systeemi käsitellään vain silloin, kun jokin tapahtuma esiintyy riippumatta tapahtumien välillä kuluneesta ajasta.

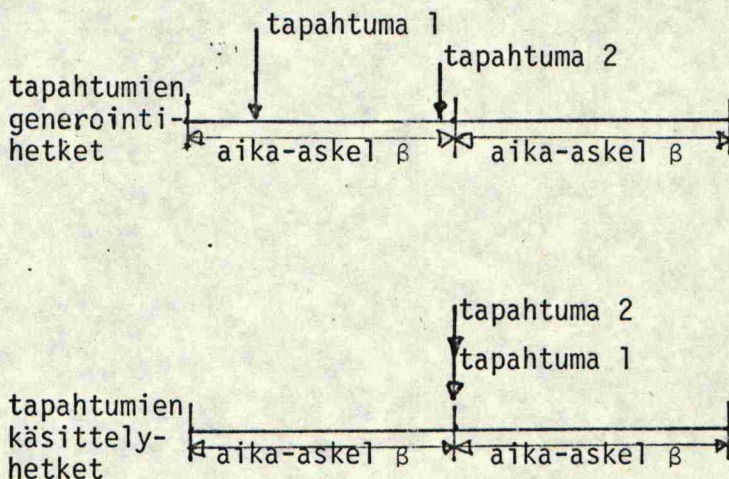
Koska aika-askel-menetelmässä tapahtuman generoinnin ja tapahtuman esiintymisen välillä ei ole välttämättä muuta korrelaatiota kuin se, että edellinen tulee ensimmäiseksi, ohjelma käsittelee kaikki aikavälillä esiintyneet tapahtumat vasta tietyllä ajanhetkellä. Tätä menetelmää käytettäessä esiintyy kahdentyyppisiä virheitä:

1. Täsmällisyyttä menetetään siksi, että tapahtumia ei aina käsitellä niiden esiintymisjärjestyksessä
2. Tarkkuus pienenee, koska tapahtuman esiintymisen simu-

lointiaika on esitettävissä vain aika-askelen tarkkuudella. Kuvat 2-4 ja 2-5 havainnollistavat virheiden syntymistä.



Kuva 2 - 4. Esimerkki tarkkuuden menetyksestä.



Kuva 2 - 5. Esimerkki täsmällisyyden menetyksestä.

Tämän tyyppisten virheiden pienentämiseksi aika-askelien olisi oltava lyhyitä, jolloin todennäköisyys, että saman tilanteen useita toisiaan seuraavia tapahtumia käsitellään yhdessä hetkessä, pienenee. Lyhyet aika-askeleet kuitenkin kasvattavat sen todennäköisyyttä, että aikavälillä ei tapahtuisi mitään, jolloin tietokoneajoaika kasvaa. Aika-askel-menetelmää käytettäessä ovat tietokoneen tehokkaan käytön ja em. kahden virhelähteen vaatimukset ristiriitaisia.

Tilanne palautuu tyypilliseen optimitoimintaehto-
etsimiseen. Kohdassa 3.5.2 käsitellään lisäksi ete-
nemismallin asettamia vaatimuksia aika-askelen pituuden
valitsemisessa.

Tapahtuma-askel-menetelmässä em. ristiriitaisia vaati-
muksia ei esiinny, sillä tapahtumat esiintyvät generoi-
mishetkinään. Jos kuitenkin systeemissä on paljon ta-
pahtumia, niiden käsittely kunkin syntymishetkellä vaa-
tisi koko systeemimallin käsitlemistä vieläkin tiheäm-
min kuin aika-askel-menetelmässä, jolloin vastaavasti
koneajan kulutus kasvaisi. Liikennesimulointeja ajatel-
len valitsemalla aika-askel sopivan pituiseksi saadaan
simuloinnin tarkkuus ja täsmällisyys riittäväksi. Vali-
tettavasti suoritetuista liikennesimuloinneista ei ole
tehty tutkimuksia ajansiirtomekanismin vaikutuksesta tie-
tokoneajokaikoihin. Useimmat referoidut liikennesimuloin-
nit on suoritettu käyttäen aika-askel-menetelmää tasaisin
aikavälein. Tapahtuma-askel-menetelmän käytöstä on vain
yksi raportti /21/, jonka mukaan sen tietokoneen ajoaika
oli huomattavasti lyhyempi kuin tasa-aika-askel-menetel-
missä, mutta simuloitava risteyskin oli yksinkertainen.
Toisessa tapahtuma-aika-simuloinnissa /23/ päädyttiin
samanlaisiin aikalukemiin kuin aika-askel-menetelmiä
käyttäen. Yleisesti tarkastellen aika-askel-menetelmä
on suoraviivaisempi ja yksinkertaisempi ymmärtää ja oh-
jelmoida. Tässä mallissa käytetään tasa-aika-askel-mene-
telmää.

2.2.4

Simuloinnin kustannuksista

Vaikka taulukon 2-1 mukaan simuloinnin kustannukset muihin
systeemin tutkimusmenetelmiin verrattuna eivät olekaan suu-
ret, voidaan niitä varsinaisten tuotantoajojen kohdalla
pienentää valitsemalla ajojen suorittamiseen sopivat tie-
tokoneet ja valitsemalla mallin parametrit koesuunnittelua
ja varianssin pienennys -menetelmää hyväksikäyttäen.

Jälkimmäisiä vaihtoehtoja käsitellään tarkemmin 5. luvussa.

Simulointiajot tietokoneella ovat tyypillisesti runsaasti keskusyksikköaikaa kuluttavia pitkien, toistuvien ja numeropyöritystä vaativien aritmeettisten ja loogisten laskutoimitusten vuoksi. Tällaiset probleemat ovat useinkin liikaa aikaa kuluttavia ajettavaksi suurtietokoneilla ennen muuta kustannuksia tarkasteltaessa.

Artikkelissa /22/ on vertailu 15-Ksanaisen (32-bittisen) simulointiohjelman toteuttamista ja ajamista pien- ja suurtietokoneilla. Suurehkojenkin ohjelmien kääntäminen ja implementointi on suoritettavissa pienoistietokoneella segmenttoimalla ohjelmat, jolloin keskusmuistissa tarvitsisi olla vain kulloinkin käsiteltävä ohjelmasegmentti. Tällä tavoin toimimalla pientietokoneella voidaan ajaa pitkiäkin ohjelmia vieläpä suhteellisen nopeasti. Artikkelista /22/ lainattuun taulukkoon 2-2 on kerätty tiedot em. esimerk kisimuloinnin vaatimat ajoajat ja aiheuttamat kustannukset eri tietokonesysteemejä käytettäessä. Vaikka taulukon absoluuttiset arvot eivät tarkasti vastaisikaan todellisuutta, ovat kokonaiskustannusten suhteelliset erot (80...100 kertaiset) siksi suuret pientietokoneiden hyväksi, että kannattaa vakavasti harkita pientietokoneen käyttämistä tämänkin liikennesimuloinnin tuotantoajojen suorittamisessa. Simulaattorin ohjelmointi, testausajot ja implementointi on nyt suoritettu UNIVAC 1108 suurtietokoneella.

Taulukko 2 - 2. Vertailu suhteellisista simulointi-
kustannuksista käytettäessä eri tieto-
konejärjestelmiä.

Company & Model	Core Size Words (k)	Cost (\$/Hr)	Run Time (Sec)	Speed Ratio	Normalized Cost, for computational equivalent of 1 hr run on the IBM 360/75 (\$/Hr)	Cost Ratio relative to the 16k NOVA 800 with F/P hardware
IBM						
360/75	262	500.	20.339	1.000	500.00	82.1
360/67	262	500.	33.056	1.625	812.50	133.4
CONTROL DATA: 6600	131	500.	8.808	0.433	216.50	35.5
SPERRY-RAND: Univac	262	500.	22.470	1.105	552.50	90.7
DIGITAL EQUIPMENT System 10/Model 1070	96	250.	43.186	2.123	530.75	87.1
DATA GENERAL Nova 800 with floating point hardware	12	1.472	77.967	3.833	5.642	0.926
	16	1.626	76.187	3.746	6.091	1.000
	20	1.779	76.627	3.571	6.353	1.0
	24	1.933	68.717	3.379	6.532	1.07
	32	2.241	52.817	2.597	5.820	0.955
Nova 800 with floating point software	16	1.318	280.820	13.807	18.198	3.0

3. SIMULOINTIMALLIN KONSTRUOINTI

Kehitetyn digitaalisen simulointimallin konstruointi toteutettiin kahdessa vaiheessa. Ensiksi systeemin osatointoja ja komponentteja kuvaavat mallit konstruointiin matemaattisin yhtälöin. Toiseksi ohjelmoitiin osamalleja dynaamisesti prosessoiva tietokoneohjelma, joka syntetisoi systeemin tilan muutokset.

Kehitetyn simulaattorin ensimmäinen toteutusvaihe kuvataan 3. luvussa ja mallin ohjelmointi 4. luvussa. Nämä luvut muodostavat simulaattorin atk-suunnittelun ja -toteutuksen varsinaisen dokumentin, jonka käsittelyjärjestys noudattaa soveltuvien osien VTT:n atk-palvelutoimistossa kokeilukäytössä olevan teknis-matemaattisten ohjelmien dokumentointistandardin mukaista jaotusta.

3.1 Simulointimallin rakenne ja sisäinen kirjanpito

Simulaattorisysteemin sisäinen kirjanpito on keskeisimpiä piirteitä liikennesimulointimalleissa. Jotta simulointiohjelma toimisi oikein, on tärkeätä, että on tehokas kirjanpitoproseduuri, jolla ei vain kuvata liikennevirtaa tietokoneessa ja pidetä kirjaa systeemin ajoneuvoista, vaan jolla nämä tehtävät suoritetaan tietokonemuistin ja ajoajan suhteen optimaalisesti.

Ajoneuvovirta voidaan esittää digitaalisessa simuloinnissa kahdella selvästi toisistaan eroavalla tavalla: ns. fysiikaalista esitystapaa tai ns. muistiesitystapaa käyttäen. Ensimmäisessä menetelmässä binäärilukuja 1 ja 0 käytetään esittämään ajoneuvoja ja niiden sijaintia. Tutkittavan liikennesysteemin geometrisen muodon mukaan voidaan tietokonemuisti organisoida, jolloin ajoneuvot sijaitsevat binäärilukuja 1 vastaavissa paikoissa. Sopivilla algebrallisilla laskutoimituksilla 1-bitit saadaan liikkumaan ja siten kuvaamaan ajoneuvojen liikettä. Fysikaalinen esitystapa on

hidas ja monimutkainen kompleksien risteys­simuloinnin suorituksessa.

Toisessa menetelmässä, muistiesitystavassa, tarvittava tieto talletetaan yhteen tai useampaan tietokonesanaan, jota voidaan käsitellä sopivilla tietokonerutiineilla. Tietoina voivat olla esimerkiksi ajoneuvojen sijainnit, nopeudet, kiihtyvyydet, pituudet ja tavoitenopeudet. Tämän menetelmän heikkous on se, että simuloitavassa systeemissä etäisyys on ilmaistavissa vain diskreetisti käyttäen yksikköblokkeja, joiden on oltava keskimääräisen ajoneuvopituuden kerrannaisosia. Tällöin ajoneuvon varaama tila on tietty lukumäärä diskreettejä paikkapositioita.

Modifoidussa muistiesitystavassa matemaattisia liikeyhtälöitä käytetään joka aika-askel ajoneuvon uuden sijainnin laskemisessa entisen sijainnin, nopeuden ja kiihtyvyyden perusteella. Tätä tekniikkaa käyttäen ajoneuvon sijainti on jatkuva suure valitun aika-askelen tarkkuudella. Muistiesitystavoiissa ajoneuvot ovat yksilöitävissä ja niiden karakteristiket ja operatiiviset suureet ovat saatavissa selville.

Käyttämällä modifoidun muistiesitystavan lisäksi ajoneuvojen identifioinnissa ketjutustekniikkaa ajoneuvojen ominaisuudet ovat helposti saatavissa selville. Tässä menetelmässä ajoneuvoille annetaan yksikäsitteinen indeksi ja ajoneuvon ominaisuudet talletetaan taulukoihin, jolloin tietyn ajoneuvon ominaisuudet saadaan aina selville ao. taulukosta ajoneuvoa vastaavan indeksin osoittamasta kohdasta. Kullakin ajoneuvolla on lisäksi alkionsa kahdessa taulukossa, joissa ko. ajoneuvon edessä ja takana olevien ajoneuvojen indeksit ovat. Tällä tavoin ajoneuvojen käsittely on yksinkertaista ja tietokoneajoaika säästyy.

3.2

Simuloinnin aika-askelen valinta

Kuten kohdassa 2.2.3 mainittiin, tässä työssä käytetty ajan-ajansiirtomekanismi on tasa-aika-askelmenetelmä. Digitaalisen tietojenkäsittelyn diskreettisen luonteen vuoksi systeemin kaikkien ajoneuvojen liikkeisiin liittyviä päätöksiä ei voida suorittaa samanaikaisesti. Koska tietokone voi tehdä vain yhden loogisen toiminnon kerrallaan, systeemin kaikkiin suureisiin liittyvät päätökset käsitellään peräkkäin tietyin aikavälein.

Tämän aika-askelen valinta on kaikkien simulointimallien kohdalla tärkeä aspekti. Jos aika-askel on liian pitkä, simulaattori menettää täsmällisyyttään sen vuoksi, että monet tapahtumat voivat jäädä käsittelemättä. Jos taas aika-askel on liian lyhyt, tietokoneajoaika kasvaa turhaan. Lisärajoituksia aika-askelen valinnalle asettavat ajoneuvojen etenemistä kuvaavat yhtälöt, joissa esiintyy viivettä kuten kohdassa 3.5.2 esitetään. Useissa referoiduissa simulointimalleissa yhden sekunnin aika-askel on havaittu tyydyttäväksi, mutta jos reaktioviive oletetaan malliin mukaan, on käytettävä lyhyempää aika-askelta. Tämän mallin testiajoissa aika-askeleena käytettiin 0,5 sekuntia. Koska aika-askelen pituus on jätetty malliin valittavaksi parametriksi, voidaan mallin kalibroinnin jälkeen aika-askelpituuden optimaalista valintaa tutkia lisää kohdassa 2.2.3 esitettyjen periaatteiden puitteissa.

3.3 Katuristeyksen rakenteen esittäminen

3.3.1 Määritelmiä

Havainnollisuuden vuoksi tarkastellaan kuvan 3-1 esittämää risteystä. Risteys ajatellaan rakentuvaksi ns. ajourista, jotka on määritelty sellaisiksi ajokaistan osiksi, joissa liikennevirta aina välttämättä pysyy muuttumattomana. Ne ovat siis haarautumis- ja liittymispisteiden välisiä ajokaistan osia.

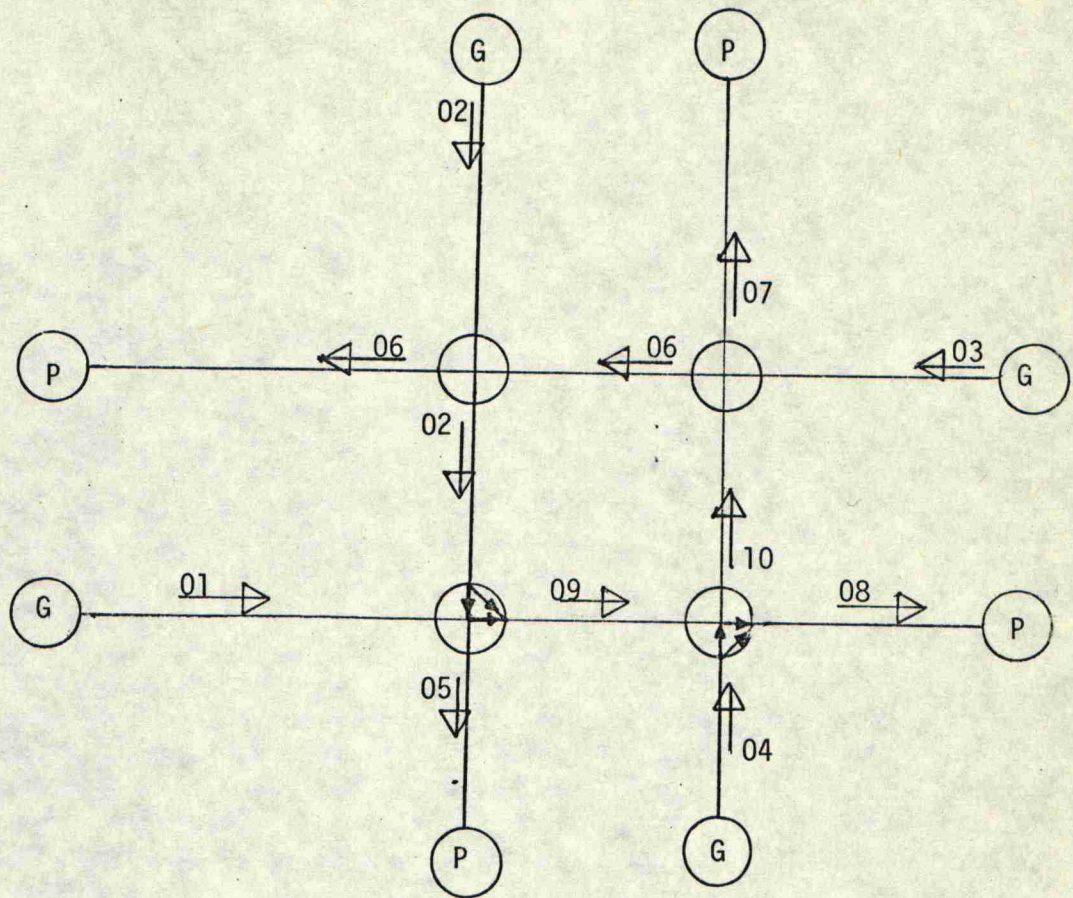
Haarautumispisteellä tarkoitetaan risteuksen sellaista kohtaa, jossa liikennevirta haarautuu kahdeksi eri liikennevirraksi. Liittymispisteellä tarkoitetaan risteuksen sellaista kohtaa, jossa kaksi liikennevirtaa liittyy yhdeksi liikennevirraksi. Vain ajourien alussa tai lopussa kaksi ajoneuvovirtaa voi haarautua tai liittyä toisiinsa.

Niitä ajouria, joita pitkin ajoneuvot saapuvat risteykseen, kutsutaan sisääntulokaistoiksi, joiden alkupäissä ovat ajoneuvojen generointipisteet. Vastaavasti risteysalueen jälkeisiä ajouria kutsutaan ulosmenokaistoiksi, joiden lopussa ovat ajoneuvojen poistumispisteet.

Ajourat yhdistetään ajoreiteiksi ketjuttamalla ne ns. rakennematriisien avulla, jolloin risteuksen geometrista muotoa ja sallittuja reittimahdollisuuksia on helppo muunnella syöttötietoja muuttamalla. Rakennematriiseilla saadaan fyysinen risteys esitetyksi tietokonekoodeina.

Risteyksessä kahden ajouran välinen kohtauspiste, joka voi olla joko ajourien leikkauspiste tai liittymispiste muttei haarautumispiste, määritellään konfliktipisteeksi.

Simulointimallin liikkuvina elementteinä on ajoneuvoja, jotka ovat jaettavissa viiteen eri luokkaan (voidaan konkretisoida esimerkiksi henkilöautoiksi, busseiksi, 1-osaisiksi ja 2-osaisiksi raitiovaunuiksi sekä polkupyöriksi).




ajoura - esim. 10


ajoreitti - esim. 04-10-07


sisääntulokaistat - 01...04


ulosmenokaistat - 05...08

 = konfliktipiste

 = generointipiste

 = poistumispiste

 = haarautumispiste

 = liittymispiste

Kuva 3 - 1. Katuristeyksen rakenteen kooditus.

Jalankulkijat eivät ole mallissa varsinaisina aktiivisina elementteinä, vaan ne käsitellään kollektiivisesti tietyllä tavalla liikkuvana todennäköisyysmassana.

3.3.2

Koordinaatisto

Risteysmalliin sovitetaan sellainen yksidimensioinen koordinaatistosysteemi, jossa paikan määrittää koordinaatti yhdessä ajouran identifiointinumeron kanssa. Itse asiassa koordinaatistoa voidaan pitää myös kaksidimensioisena, kun toisella koordinaatilla ilmaistaan jatkuvaa suuretta: pituusyksiköitä ja toisella diskreettiä suuretta: ajouran numeroita. Yksidimensioisen koordinaatistosysteemin origot asetetaan ajourien alkupisteisiin.

Aina ajoneuvon vaihtaessa ajouraa se siirtyy uuteen koordinaatistoon eteneepä se sitten suoraan tai kääntyen. Valittu koordinaatistosysteemi soveltuu yhtä hyvin suorien kuin kaarvienkin ajourien kuvaamiseen.

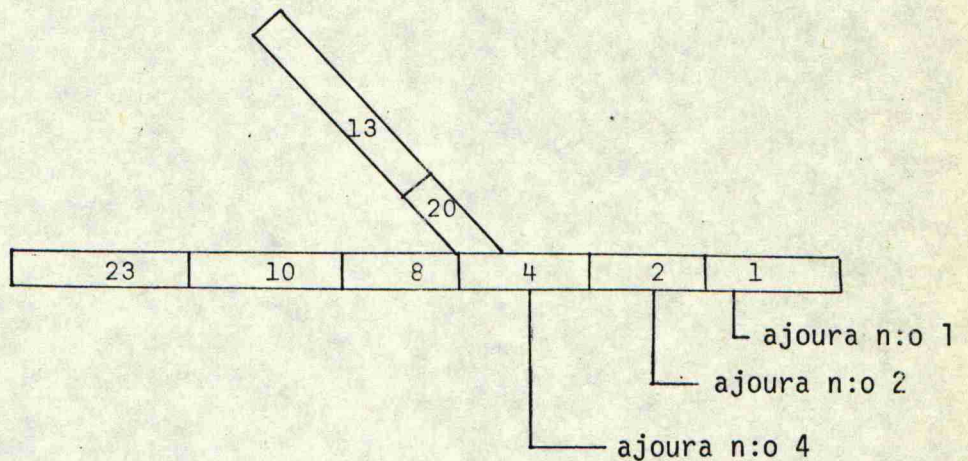
3.3.3

Risteyksen käsittelytekniikka

Kukin ajoura identifioidaan yksikäsitteisellä indeksillä. Ajouran suhteellinen asema risteysysteemin muihin ajouriin nähden on loogisesti organisoitu ketjutustekniikalla (kuva 3-2). Kuhunkin ajouraan liitetään kaksi arvoa kahdella rakennematriisilla, joissa ovat heti ajouran edessä ja sen jäljessä olevien ajourien indeksit tai jos jonkin ajouran edestä lähtee kaksi ajouraa (haarautumispiste), molempien ajourien indeksit ja vastaavasti liittymispisteessä, jolloin ajouralla on kaksi jäljessä olevaa ajouraa. Taulukoiden nimet ohjelmassa ovat vastaavasti EURA ja SURA. Tällä tavoin rakennettua risteystä käsiteltäessä päästään helposti siirtymään ajouria pitkin eteenpäin tai taaksepäin tarpeen mukaan.

Ajoneuvojen kuljettamiseksi risteysksen läpi kukin ajoneuvo varustetaan yksikäsitteisellä identifiointikoodilla, jonka se on saanut generoinnissa.

Indeksi	Edeltäjäajoura (EURA)		Seuraaja-ajoura (SURA)	
	suoraan	kääntyen	suoraan	kääntyen
1	0	0	2	0
2	1	0	4	0
4	2	0	8	20
8	4	0	10	0
10	8	0	23	0
13	20	0	0	0
20	0	4	13	0
23	10	0	0	0



Kuva 3 - 2. Ketjutustekniikan käyttö fyysisen ajourajonon esittämisessä.

Ajoneuvojen käsittelyn organisointi on suoritettu myös ketjutustekniikalla aivan identtisellä tavalla ajourien käsittelytekniikan kanssa. Ainoastaan ajoneuvojen identifiointikoodin valinta on poikkeava. Identifiointikoodi on yksikäsitteinen tietyillä peräkkäisillä ajanhetkillä, mutta saman simulointiajon aikana samaa identifiointikoodia voi käyttää useampikin ajoneuvo.

Konfliktipisteet identifioidaan yksikäsitteisillä indekseillä ja ne paikallistetaan kohtaavien ajourien alkupäistä mitattujen etäisyyksien avulla.

3.4

Ajoneuvojen esittäminen

3.4.1

Ajoneuvoihin ja ajajiin liittyvät ominaisuudet

Mallissa ajoneuvon ominaisuudet esitetään seuraavan 12 tietosanan joukkona, joka sisältää riittävät tiedot ajoneuvon kuljettamiseksi systeemin läpi (merkitsemätön indeksi on ajoneuvon id-koodi):

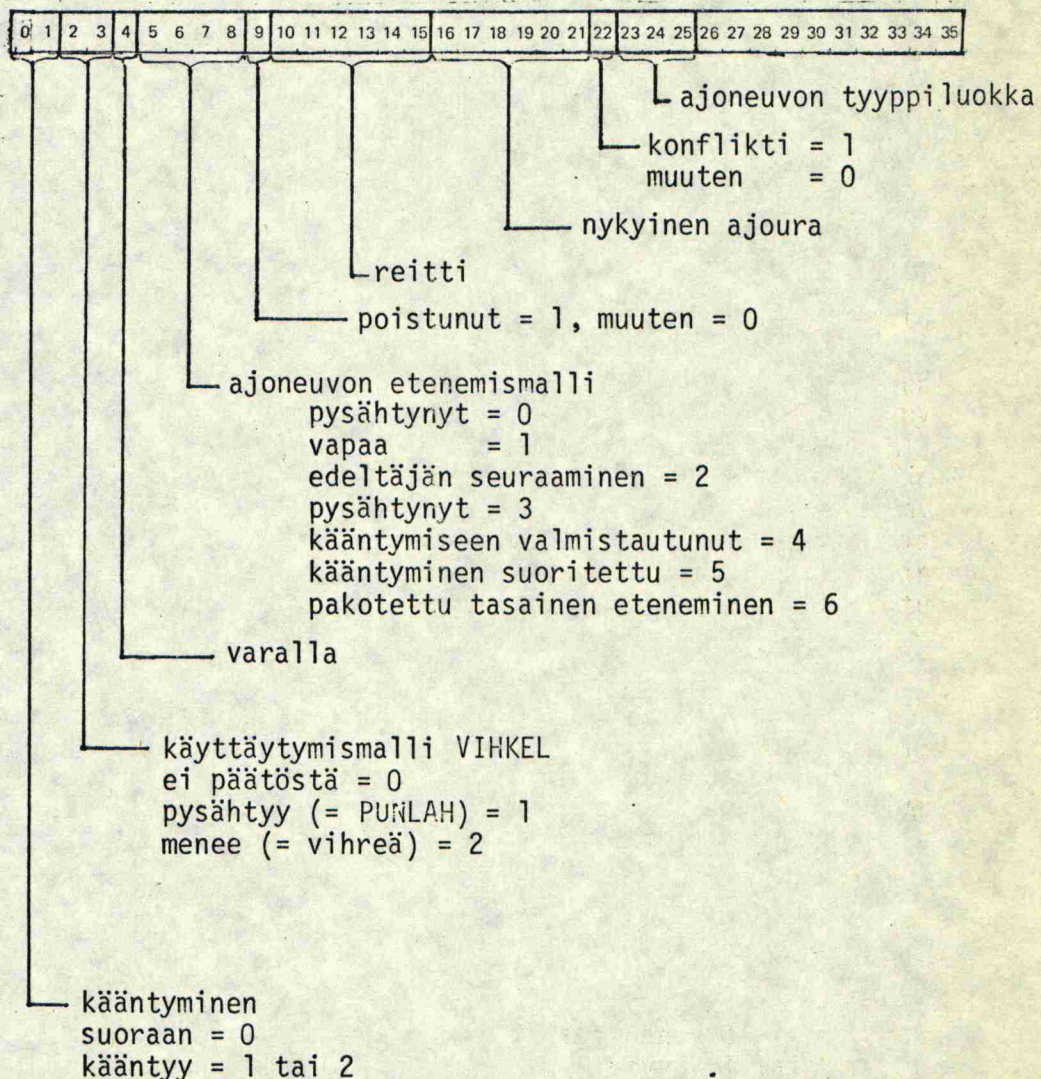
1. paikka, x (.)
2. nopeus, v (.)
3. kiihtyvyys, A (.,IT)
4. pakattu sana, AUOMI (.)
5. systeemiin tuloaika, AU (.,1)
6. tavoitenopeus, AU (.,2)
7. haluttu jarrutushidastuvuus AU (.,3)
8. maksimikiihtyvyys, AU (.,4)
9. efektiivinen pituus, AU (.,5)
10. pysähtymispaikka, XPYS (.)
11. seisona-aika, TPYS (.)
12. karakteristinen nopeus, ALFA (.)

Tiedot 1, 2, 3 ja 4 ovat dynaamisia ja ovat muutettavissa koko simuloinnin ajan joka aika-askel.

Ajoneuvon paikka ilmaistaan sen ajouran alkupään suhteen, jolla ajoneuvo kulloinkin on. Ajoneuvon suhteen paikkakoordinaatti on sijoitettu ajoneuvon etuosaan.

Nopeus (tieto2) on se todellinen nopeus, joka ajoneuvolla on tietyllä ajanhetkellä kun muun liikenteen ja ympäristön asettamat rajoitukset on otettu huomioon. Ajoneuvon ja sen ympäristön tilasta riippuva kiihtyvyysarvo vaikuttaa ajoneuvon käyttäytymiseen vasta ajajan reaktioajan kuluttua. Kun reaktioaika on simuloinnin aika-askelta pidempi, on aika-askeleittain lasketut kiihtyvyysarvot talletettava viiveenä esiintyvän reaktioajan verran. Mallissa kiihtyvyydeksi määritelty suure voi saada sekä positiivisia että negatiivisia arvoja. Joissakin yhteyksissä negatiivisesta kiihtyvyydestä käytetään nimitystä hidastuvuus.

Tietosanaa 4 on jaettu eri pituisiksi kentiksi ja siihen on pakattu useita tietoja. Viittaukset siihen tietojen poistamiseksi, käsittelemiseksi tai uudelleen tallettamiseksi tapahtuvat tietokonesanan käsittelyyn kehitetyllä FORTRAN V:n sisäisellä funktiolla FLD. Tietosanan 4 sisältönä on kuvassa 3-3 esitetyt ajoneuvokohtaiset ominaisuudet.



Kuva 3 - 3. Ajoneuvokohtaisten tietojen sijoittaminen tiivistettyyn sanaan.

Tiedot 5, 6, 7, 8 ja 9 ovat vakioita kullekin ajoneuvolle koko simulointikäsittelyn ajan ja ne valitaan satunnaisprosesseilla jakautumista, joiden parametrit on luettu syöttötietoina.

Tieto 5 on simulointikellon arvo hetkellä, jolloin ajoneuvo generoidaan sisääntulokaistalleen. Tiedon 6, tavoitenopeuden valinta on muuten edellisten kaltainen, mutta sillä on muuttuva jakautumaparametri, joka riippuu ajoneuvon sijainnista ajourien suhteen. Tieto 7 on haluttu jarrutushidastuvuus, jota ajoneuvo pyrkii käyttämään, kun se alkaa pysähtyä vakiohidastuvuudella tiettyyn ennalta laskettuun pysähtymispaikkaan mennessä. Jollei ajoneuvo ole pakotettu käyttämään nopeampaa hidastuvuutta, esimerkiksi keltaisen liikennevalon vuoksi, se noudattaa pysähtymismallissaan haluttua jarrutushidastuvuutta tai vähän sitä nopeampaa hidastuvuutta. Halutulla jarrutushidastuvuudella kuvataan sitä rajahidastuvuutta, jota nopeamat jarrutukset tapahtuvat pysähtymismallin mukaan. Jos ajoneuvon hidastuvuus on haluttua jarrutushidastuvuutta nopeampi (pysähtymismalli), jolloin myös sen jarruvalojen voidaan katsoa palavan, alkaa myös seuraaja-ajoneuvo pysähtyä. Ajoneuvon sovellettavalla etenemismallilla laskettua kiihtyvyyssarvoa verrataan maksimikiihtyvyyteen (tieto 8) ja niistä valitaan pienin. Tieto 9 efektiivinen pituus muodostuu ajoneuvon todellisesta pituudesta ja sen vaatimasta minimivälimatkasta seuraavaan ajoneuvon. Ajoneuvon todellisena pituutena pidetään 80 % efektiivisestä pituudesta. Pysähtymismalleissa käytetään yleensä efektiivistä pituutta, mutta tiukassa tilanteessa käytetään todellista pituutta. Kullekin sisääntulokaistalle voidaan ajoneuvoja generoida kolmesta eri-keskiarvoisesta efektiivisen pituuden jakautumasta, joiden suhteelliset osuudet on luettu syöttötietoina.

Joka kerta kun ajoneuvo aikoo pysähtyä, määritetään pysähtymispaikan koordinaatti. Tieto 10 voi siis muuttua kerran tai useammin ajoneuvon kulkiessa systeemin läpi tai jos ajoneuvon ei kertaakaan tarvitse pysähtyä ei tietoa 10 määritetä ollenkaan. Tieto 11 on ajan suhteen kumulatiivinen tilastotieto, joka ilmaisee ajan, jonka ajoneuvo on joutunut olemaan pysähtyneenä kulkiessaan systeemin läpi. Tieto 12 liittyy edeltäjän seuraamismalliin, jossa sitä käytetään ajoneuvojen reaktioherkkyysteki-

jän parametrina. Se on ajoneuvokohtainen ja se valitaan normaalijakautuneesta jakautumasta.

3.4.2

Ajoneuvojen identifiointi

Ajoneuvojen kuljettamiseksi risteysysteemin läpi kukin ajoneuvo varustetaan yksikäsitteisellä indeksillä, jonka se on saanut generointihetkellä. Organisoimalla käsittelysysteemi loogisesti ketjutustekniikkaa (vrt. kuva 3-4) hyväksikäyttäen saadaan selville ajoneuvon suhteellinen asema toisiinsa nähden eli ajoneuvon edessä ja jäljessä olevien ajoneuvojen indeksit. Nämä ajoneuvokohtaiset tiedot ovat taulukoissa EAUTO (edeltäjäajoneuvo) ja SAUTO (seuraaja-ajoneuvo). Täten on mahdollista löytää nopeasti viereisten ajoneuvojen ominaisuudet ja saada risteyksessä olevien ajoneuvojen sijainnista kokonaiskuva. Menetelmä on samantapainen kuin ajourien ketjutuksessakin käytetty (kohta 3.3.3) paitsi se, että taulukon alkioissa nyt on vain yksi identifiointi-indeksi.

Ajoneuvon indeksi	Edeltäjäajoneuvo (EAUTO)	Seuraaja-ajoneuvo (SAUTO)
9	0	20
20	9	31
31	20	32
32	31	46
46	32	0



Kuva 3 - 4. Ketjutustekniikan käyttö ajoneuvojen esittämisessä.

Käsittelyn nopeuttamiseksi kullakin ajouralla on alkionsa taulukoissa EAURA (ensimmäinen ajoneuvo uralla) ja VAURA (viimeinen ajoneuvo uralla), joista edellinen sisältää ajouralla en-

simmäisenä olevan ajoneuvon identifiointi-indeksin ja jälkimmäinen vastaavasti ajouralla viimeisenä olevan ajoneuvon indeksin. Koska koko risteyksen käsittely tapahtuu hierarkisesti ajourittain taulukon SURA osoittamassa järjestyksessä ja kullakin ajouralla olevien ajoneuvojen käsittely taulukon SAUTO osoittamassa järjestyksessä, niin taulukoiden EAURA ja VAURA tiedoilla päästään näppärästi käsiksi taulukon SAUTO oikeaan alkioon.

Taulukko 3 - 1. Päätöstaulu ajoneuvon identifiointitaulukoihin tehtävistä muutoksista ajoneuvon siirtyessä uuteen ajouraan.

Taulukossa käytetään indekseinä seuraavia lyhenteitä:

KO = käsiteltävänä oleva ajoneuvo

EA = edeltäjäajoneuvo

SA = seuraaja-ajoneuvo

VU = ajoura, jolla käsiteltävänä oleva ajoneuvo oli ennen siirtymistä

EU = suoraan VU:n edessä oleva ajoura

UU = uusi ajoura, jolle käsiteltävänä oleva ajoneuvo on juuri siirtynyt

Päätöstaulusssa olevat muutosvaihtoehdot:

M = vastaava tieto muuttuu

E = vastaava tieto ei muutu

E* = vastaavan tiedon ollessa = 0, se muuttuu, muuten ei muutu

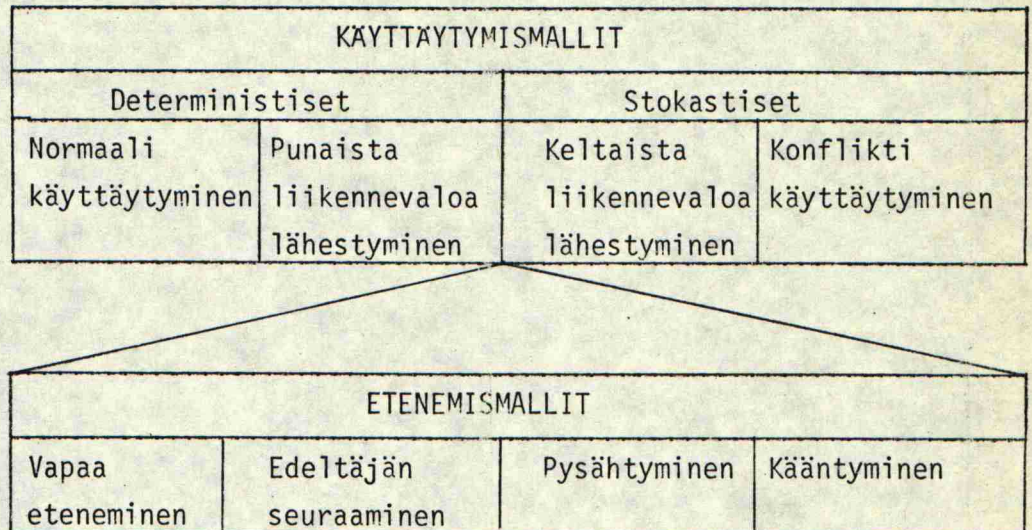
- = ko. tiedon tarkastelu on irrelevanttia

TAPAHTUMA	EAUTO (KO)	SAUTO (KO)	EAUTO (EA)	SAUTO (EA)	EAUTO (SA)	SAUTO (SA)	EAURA (VU)	EAURA (ED)	EAURA (UU)	VAURA (VU)	VAURA (ED)	EAURA (UU)
GENEROINTI												
	=0	=0	-	-	-	-	-	-	M	-	-	M
	≠0	=0	E	M	-	-	-	-	E	-	-	M
AJOURAN VAIHTO												
SUORAAN EDETEN												
	=0	=0	-	-	-	-	M	-	M	M	-	M
	≠0	=0	E	E	-	-	M	-	E	M	-	M
	=0	≠0	-	-	E	E	M	-	M	E	-	M
	≠0	≠0	E	E	E	E	M	-	E	E	-	M
AJOURAN VAIHTO												
KÄÄNTYEN												
	=0	=0	-	-	-	-	E*	E	M	M	E	M
	≠0	=0	E	M	-	-	M	E	E*	M	E	M
	=0	≠0	-	-	M	E	M	E	E*	E	E	M
	≠0	≠0	E	M	M	E	M	E	E*	E	E	M
POISTUMINEN												
	=0	=0	-	-	-	-	M	-	-	M	-	-
	=0	≠0	-	-	M	E	M	-	-	E	-	-

3.5

Ajoneuvojen etenemismallit

Liikenne-risteyksen simulointimallin keskeisimpiä asioita ovat ajoneuvojen dynaamiset riippuvuudet toinen toisistaan, joita kuvataan neljällä etenemismallilla, joiden soveltamista ohjaa neljä käyttäytymismallia. Ajoneuvojen välisten dynaamisten riippuvuuksien käsittely voidaan esittää hierarkisesti 2-tasoisena järjestelmänä kuvan 3-5 mukaisesti.



Kuva 3 - 5. Käyttäytymis- ja etenemismallien hierarkkinen rakenne.

Käyttäytymismallit voidaan jakaa kahteen luokkaan: deterministiset ja stokastiset. Deterministisiksi malleiksi luetaan "normaali" eteneminen ja punaista liikennevaloa lähestyminen. Normaalilla käyttäytymisellä tarkoitetaan ajoneuvon etenemistä niissä tilanteissa, joissa sen liike määräytyy sen edessä samaan suuntaan etenevän ajoneuvon käyttäytymisestä, tai jos sellaista ajoneuvoa ei ole, sen liikettä ei rajoita muu kuin sen omat ominaisuudet. Ajoneuvon lähestyessä punaista liikennevaloa sen liike määräytyy pysähtyväksi ajoneuvon edettyä tarpeeksi lähelle liikennevaloa.

Stokastisiin käyttäytymismalleihin luetaan keltaista liikennevaloa lähestyminen ja käyttäytyminen konfliktitilanteessa. Ajajat joutuvat tekemään omaa liikettään koskevia päätöksiä

annettujen todennäköisyysparametrien nojalla. Näistä päätöksentekomalleista on tarkempi kuvaus kohdassa 3.6.

Etenemismalleja on neljä: vapaa eteneminen, edeltäjän seuraaminen, pysähtyminen ja kääntyminen. Viidenneksi mainittakoon etenemismalli, jossa ajoneuvon nopeus on nolla ja se on siis pysähtynyt. Triviaalisuutensa vuoksi pysähtynyt -mallia ei seuraavassa käsitellä enempää. Käyttäytymismalleilla valitaan kunkin ajoneuvon kohdalla noudatettava etenemismalli ja päätetään siirtymisestä jonkin toisen etenemismallin piiriin.

Kukin etenemismalli on kuvattavissa yhtälöllä, joita tarkastellaan kohdissa 3.5.1...3.5.5. Etenemismalleissa käytetyt yhtälöt antavat vain ajoneuvon kiihtyvyysarvot (+ tai -), joita aletaan noudattaa reaktioviiveen T kuluttua ja joita noudatetaan simuloinnin aika-askelen c ajan. Digitaalisessa simuloinnissa T on oltava jokin aika-askelen moninkerta. Vapaan etenemismallin käsittelyn yhteydessä johdetaan liikeyhtälöt, joiden avulla simuloinnin aika-askelen pituiset ajoneuvojen liikkeet lasketaan. Näitä paikan ja nopeuden liikeyhtälöitä voidaan soveltaa myös muiden etenemismallien yhtälöiden antamiin kiihtyvyysarvoihin.

3.5.1

Vapaa eteneminen

Mallissa ajajan oletetaan pyrkivän johonkin tavoitenopeuteen aina, kun hänellä on mahdollisuus siihen. Tällöin ajoneuvon sanotaan noudattavan vapaan etenemisen mallia.

Jonon ensimmäisen ja muiden vapaasti etenevien ajoneuvojen kiihtyvyyksien oletetaan noudattavan Pipesin ja Seddonin /35/ esittämää eksponenttaalista riippuvuutta muodoltaan

$$a(t) = c V e^{-ct}, \quad (3-1),$$

jossa $a(t)$ on hetkellinen kiihtyvyys,

V on loppu- tai tavoitenopeus,

c on vakio, jonka dimensio on sek⁻¹,

e on Neperin logaritmin kantaluku.

Jos yhtälö 3-1 integroidaan, saadaan nopeudeksi:

$$v(t) = V(1 - e^{-ct}). \quad (3-2)$$

Integroimalla uudestaan saadaan

$$x(t) = V(ct + e^{-ct}) / c. \quad (3-3)$$

Olkoon kiihtyvyys hetkellä $t = 0$ alkukiihtyvyys A , jolloin yhtälöä 3-2 hyväksikäyttäen voidaan kiihtyvyys esittää muodossa

$$a(t) = A(1 - v(t) / V), \quad (3-4)$$

jossa se on lineaarisesti nopeudesta riippuva. Seddon /35/ on testannut tämän mallin autoyhtiöiltä saamillaan datoilla ja sopivilla V :n ja A :n valinnoilla malli on hänen mukaansa kelvollinen.

Jotta yhtälöitä 3-1...3-4 voitaisiin käyttää diskreetissä simulointimallissa, on ne muutettava eksponentiaalisen mallin approksimatiiviseen muotoon:

$$x(t) = x(t - c) + v(t - c) c + \frac{1}{2} a(t - c) c^2 \quad (3-5)$$

$$v(t) = v(t - c) + a(t - c) c \quad (3-6)$$

$$a(t) = A(1 - v(t) / V) \quad (3-7)$$

jossa c on aika-askel. Approksimoinnissa joudutaan oletta-
maan, että paikka, nopeus ja kiihtyvyys pysyvät vakioina koko
aika-askelen ajan, mikä todellisuudessa ei kuitenkaan pidä
paikkaansa. Kuitenkin kohdassa 3.5.2 esille tulevan dynaami-
sen stabiilisuuden vuoksi aika-askel joudutaan pitämään mel-
ko pienenä ja siksi syntyvät approksimointivirheet ovat merki-
tyksettömiä.

Vapaan etenemismallin käyttö edellyttää kullekin ajoneuvolle

kahta parametriä: tavoitenopeus V ja alkukiihtyvyys A . Sinhan /36/ raporttoimien datojen mukaan tavoitenopeutta voidaan pitää normaalijakautuneena suureena. Jakautuman parametrejä valittaessa on kuitenkin otettava huomioon se, että yleensä ajaja aliarvioi oman nopeutensa, kuten Häkkisen tutkimuksesta ilmenee /17/. Alkukiihtyvyyden jakautumasta ei kuitenkaan ole löytynyt tutkimuksia. Useimmissa simulointimalleissa se on oletettu vakioksi. Seddon olettaa intuitiivisesti jonosimuloinnissaan /34/ sen olevan normaalijakautunut. Tässä mallissa alkukiihtyvyys oletetaan vakioksi.

3.5.2

Edeltäjän seuraaminen

Koko laajuudessaan risteyskäyttäytyminen on paljolti selitettävissä seuraamisprosessien avulla. Erityisen tärkeä esimerkki tämän tyyppisestä käyttäytymisestä on jonon purkautumistapahtuma. Edeltäjän seuraamismallilla (car-following) pyritään kuvaamaan ajajan ja ajoneuvon käyttäytymistä niissä tilanteissa, joissa edellä ajavan ajoneuvon liikkeitä vaikuttavat seuraaja-ajoneuvon etenemiseen. Tämän ilmiön kuvaamiseksi on kehitetty useita malleja. Kokeellisiin ja teoreettisiin tutkimuksiin perustuvia analyttisiä tai deterministisiä teorioita on esitetty esimerkiksi artikkeleissa /4, 11, 10, 15/. Kontinuumisia ja/tai hydrodynaamisia käsitteitä on käytetty kuvaamaan jonon ajaja-ajoneuvo-ominaisuuksia artikkeleissa /30, 32/.

Edeltäjän seuraamismalli on itse asiassa erään muotoinen psykologiaan perustuva ärsyke-reaktio-yhtälö, jossa ajajan reaktio (kiihtyvyys tai jarrutus) määräytyy ärsykefunktioista, joka riippuu ajoneuvojen välisestä suhteellisesta nopeudesta ja suhteellisesta välimatkasta, absoluuttisesta nopeustasosta, ajajan herkkyydestä ja reaktioajasta sekä monesta muusta inhimillisestä, mekaanisesta ja ympäristötekijästä. Edeltäjän seuraamismalliteorian ärsyke-reaktio-yhtälöä, jotka ovat kehittäneet Herman, Potts ja Rothery /mm.15 /, käytetään myös tässä simulointimalleissa kuvaamaan risteysliikenteen tietynlaista etenemistä.

Hermanin esittämä teoria on suoraan sovellettavissa ajoneuvojen

etenemisprobleemaan digitaalisessa simuloinnissa. Kahdessa poikkeustapauksessa (kääntyminen ja pysähtyminen) edeltäjän seuraamismalli ei sovellu käytettäväksi, vaan yksityisen ajoneuvon liikkeen kuvaamiseen on käytettävä muita yhtälöitä.

Hermanin yhtälö on yleisessä muodossaan:

$$\text{reaktio} = \text{herkkyys} \times \text{ärsyke} \quad (3-8)$$

Parhaiten ajoneuvojen etenemistä kuvaava yhtälön 3-8 muotoinen yhtälö on

$$a_{n+1}(t + T) = \frac{\alpha \cdot v_{n+1}^m(t + T) \cdot [v_n(t) - v_{n+1}(t)]}{[x_n(t) - x_{n+1}(t)]^\ell} \quad (3-9)$$

jossa a_{n+1} on käsiteltävänä olevan $(n+1)$:nnen ajoneuvon kiihtyvyys hetkellä $(t + T)$

$v_n(t)$ ja

$v_{n+1}(t)$ ovat edeltäjän ja seuraajan nopeudet hetkellä t

$v_{n+1}(t + T)$ on $(n + 1)$:nnen ajoneuvon nopeus hetkellä $(t + T)$

$x_n(t)$ ja

$x_{n+1}(t)$ ovat edeltäjän ja seuraajan paikkakoordinaattien arvot hetkellä (t)

α on ajajan herkkyyskerroin (ts. reaktion voimakkuustekijä)

T on aikaviiveenä käytetty reaktioaika, joka sisältää myös havainnonteko- ja päätöksentekoajat

ℓ on eksponentti, jona tavallisesti on käytetty arvoja 0 tai 1

m on eksponentti, jona tavallisesti on käytetty arvoja 0, 1 tai 2.

Indeksillä $n + 1$ varustettu ajoneuvo seuraa indeksillä n varustettua ajoneuvoa. Indeksejä vastaavien kokonaislukujen ei välttämättä tarvitse olla peräkkäisiä (vrt. kohta 3.4.2). Joissakin edeltäjän seuraamismalleissa (esim. /11/) ajajan herkkyystekijä ja nopeus $v_{n+1}(t + T)$ on yhdistetty ns. karakteriseksi

nopeudeksi, jota käsitellään tarkemmin kohdassa 3.5.5. Käytämällä herkkyystekijänä karakteristista nopeutta välttään nopeuden $v_{n+1}(t + T)$ laskemiselta apusuureksi yhtälöön 3-9. Tässä simulointimallissa on käytetty yhdistettyä herkkyystekijää, joka on ajoneuvokohtaisesti normaalijakautunut suure. ℓ ja m saavat arvon 1. Muita ℓ :n ja m :n arvoja ovat laajimmin May ja Keller tutkineet /26/. Mallin 3-9 mukaan seuraaja-ajoneuvon kiihtyvyys reaktioajan T kuluttua on eräs funktio sen nykyisestä nopeudesta sekä nopeus- ja matkaerosta sen ja edeltäjäajoneuvon välillä. Viiveen T oletetaan sisältävän havainnonteko-, päätöksenteko- ja reaktioajan, vaikka jäljessä seuraa vassa tekstissä viivettä kutsutaankin lyhyiden vuoksi vain reaktioajaksi.

Itse asiassa ajaminen ei käytännössä tapahdu edeltäjän seuraamismallin oletusten (suhteellisen nopeuden muutos) mukaan, koska ajaja ei edes huomaa pieniä suhteellisen nopeuden muutoksia. Fox ja Lehman raportoivat /10/:ssa suhteellisen nopeuden muutoksen havaitsemiskynnyksestä. Psykologisesti tarkastellen se on silmän verkkokalvolle muodostuvan kuvan kulmaliikkeen nopeuden muutoksen saama tietty minimikynnysarvo. Kuvan 3-6 merkinnöillä kulmamutoksen nopeus $\frac{d\theta}{dt}$ voidaan määrittää yhtälöstä

$$L\theta \approx w \quad \text{eli} \quad \theta \approx \frac{w}{L} ,$$

$$\text{joten} \quad \frac{d\theta}{dt} \approx -\frac{w}{L^2} \frac{dL}{dt} ,$$

jossa w on havaittavan esineen koko 1-dimensioisena. Koska L on suhteellinen etäisyys, on

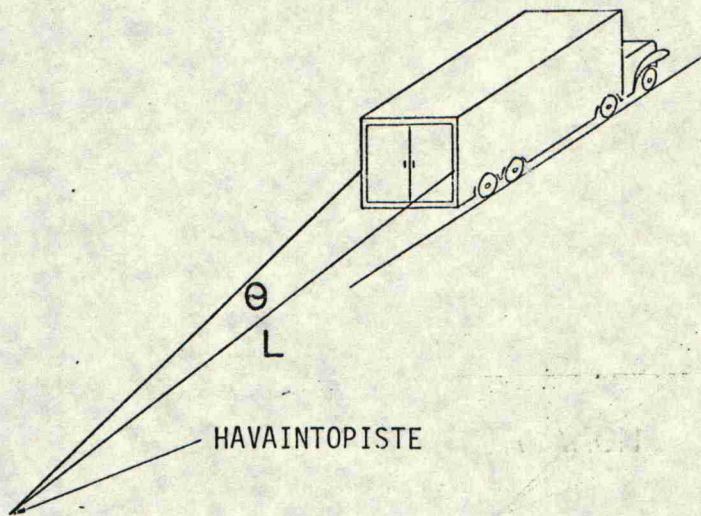
$$\left| \frac{d\theta}{dt} \right| \approx w \left| \frac{(v_n - v_{n+1})}{(x_n - x_{n+1})^2} \right| . \quad (3-10)$$

Foxin /10/ mukaan on osoitettu, että $\left| \frac{d\theta}{dt} \right|$ on suhteellisen nopeuden havaitsemisen kynnyksen alapuolella, kunnes suure

$$w \left| \frac{v_n - v_{n+1}}{(x_n - x_{n+1})^2} \right| \quad (3-11)$$

ylittää arvon $6 \cdot 10^{-4}$ /sek. Yhtälöiden 3-9 ja 3-10 vertailu osoittaa, että nämä yhtälöt ovat samanlaiset jos yhtälössä 3-9 $\ell = 2$. On mielenkiintoista havaita, että havaintokynnykseen perustuva "psykologinen" yhtälö 3-10 on yhteensopiva aivan eri perusteista johdetun empiirisen yhtälön 3-9 kanssa.

Simulointimalleissa on käytetty yhtälön 3-9 mukaista edeltäjän seuraamismallia. Ajajan käyttäytymisen katsotaan noudattavan tätä yhtälöä kuitenkin vain, jos ko. ajajalle kynnystestin arvo (yhtälö 3-11) ylittää nopeuden muutoksen havaitsemisen kynnysraja-arvon. Kun kynnystestin arvo jää pienemmäksi kuin kynnysraja-arvo, ajaja ei havaitse suhteellista nopeuseroa, vaan hänen toimintansa katsotaan tapahtuvan vapaan etenemismallin mukaan tavoitenopeuteen pyrkien.



Kuva 3 - 6. Kulmanopeuden havaitseminen.

Toinen tapaus, jossa ajoneuvo ei etene edeltäjän seuraamismallin mukaan, on se, jolloin ajoneuvon ja sen edeltäjän välinen matkaväli on syöttötietona luettua parametriä suurempi tai jolloin ajoneuvon edessä ei ole ollenkaan edeltäjäajoneuvoa. Tällöin ajoneuvo noudattaa vapaan etenemisen mallia. Esimerkiksi liikennevaloista lähtiessä jonon ensimmäisen ajoneuvon käsittely kuuluu aina tällaiseen tapaukseen.

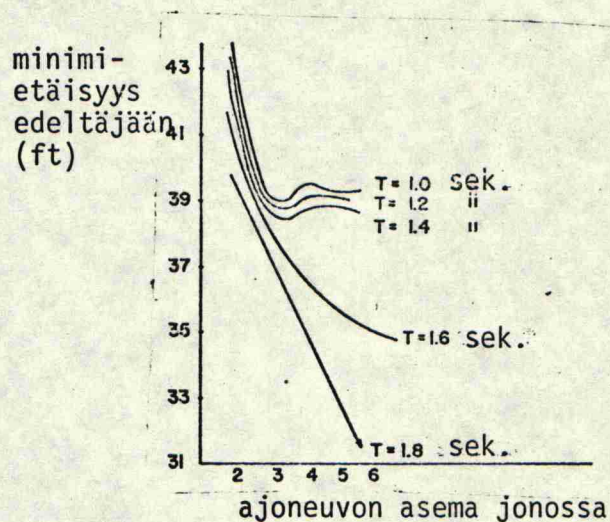
Fox ja Lehman raportoivat /10:ssa, että edeltäjäajoneuvon jarruvalojen käsittelyn mukaanotto malliin vaikuttaa stabiloivasti liikkeisiin edeltäjän seuraamismallissa, kuten se tekee todellisuudessaakin. Tässä mallissa jarruvalojen vaikutus on mukana siten, että kun edeltäjäajoneuvon liike hidastuu nopeammin kuin ajoneuvokohtainen suure: haluttu jarrutushidastuvuus (ajoneuvon moottorilla hidastaminen), jolloin edeltäjäajoneuvo noudattaa pysähtymismallia, myös käsiteltävänä olevaan ajoneuvoon sovelletaan pysähtymismallia.

Edeltäjän seuraamismallin voidaan osoittaa vaikuttavan merkittävästi ko. simulointisysteemin stabilisuusominaisuuksiin. Sitä on usein käytetty liikennesimuloinneissa, joissa aika-askel on valittu reaktioajan T suuruiseksi. Jos T on ollut esimerkiksi 1 sekunnin luokkaa, on tarvittu vakavia rajoituksia stabiilisuuden aikaansaamiseksi. Rajoituksiahan aina joudutaan käyttämään digitaalisissa simuloinneissa, mutta se, miten paljon niitä joudutaan käyttämään, voi hyvinkin olla simuloinnin onnistumiseen ja mallin tarkkuuden mittana. Vaikka edeltäjän seuraamismalli 3-9 kuvaakin tarkasti sitä tilannetta, miten ajoneuvon kiihtyvyys tai hidastuvuus T aikayksikön kuluttua on riippuvainen sen nykyisestä suhteellisesta nopeudesta, muuttuu kiihtyvyys jatkuvasti tilanteiden muuttuessa. Siksi jos simulointisysteemi käsitellään vain joka T :s aikayksikkö, samat kiihtyvyyssarvot pysyvät voimassa T ajan ja tämä voi aiheuttaa epästabiilisuutta (vrt. kohta 2.2.3). Tällöin voi myös tapahtua siten, että kun ajoneuvot ovat lähdössä stationäärisestä jonosta, ne lähtevätkin liikkeelle esimerkiksi vain joka 2 T :s sekunti, vaikka havaintojen mukaan niitten olisi lähdettävä joka T :s sekunti. Tämän

vuoksi ohjelma on kirjoitettu siten, että systeemin aika-askel on lyhyempi kuin reaktioaika. Aika-askel on valittava siten, että reaktioaika on sillä jaollinen. Aika-askelen pituus ja reaktioaika ovat ohjelmassa muutettavissa olevia parametrejä. Seddon raportoi /35/ löytäneensä yrityksen ja erehdyksen kautta parhaaksi reaktioajan ja aika-askelen suhteeksi 4 reaktioajan oïlessa 1.0 sekunti.

Fox ja Lehman raportoivat /10/:ssa edeltäjän seuraamismallin validiteettia tutkiessaan mallin helposti johtavan epästabiiliseen liikkeeseen

- a) jos suhteellisen nopeuden havaitsemiskynnys on paljon suurempi kuin $5 \cdot 10^{-4} \dots 2 \cdot 10^{-3}$ radiaania sekunnissa (,johon psykologitkin päätyivät)
- b) jos maksimijarrutuskyky asetetaan liian pieneksi, tapahtuu peräänajoja (nyt onkin tarkoitus välttää tämä piirre sallimalla suuriakin hidastuvuuksia)
- c) jos tavoitenopeuden vaihtelurajat ovat liian suuret, syntyy oskilloiva liike
- d) jos edeltäjän jarruvalot jätetään huomioon ottamatta, tapahtuu peräänajo



Kuva 3 - 7. Reaktioajan vaikutus edeltäjän seuraamismallin stabiilisuuteen.

e) jos reaktioaika on yli 1,4 sekuntia, johon useat tutkijat ovat empiirisesti päätyneet /16/, jonon käyttäytyminen on epästabiili. Kuvassa 3-7 on piirretty kahden auton minimivälimatka, kun jono, jossa ne ovat, lähtee liikkeelle ensimmäisen auton alkukiihtyvyyden ollessa $2,4 \text{ m/s}^2$. Kun reaktioaika T on pienempi kuin 1,4 sek., häiriö ei vahvistu sen edetessä jonon loppupäätä kohti, eikä yhdenkään auton välimatka edeltäjään tule n. 11 metriä pienemmäksi. Reaktioajan kasvaessa kukin autopari lähenee toisiaan ja jonon loppupäässä tapahtuu peräänajoja.

Edeltäjän seuraamismallin toiminta vaikuttaa periaatteessa oikealta. Sen käyttäytyminen sekä kvalitatiivisessa mielessä että sen yhteensopivuus /10/:n mukaan kvantitatiivisten arvojen kanssa vahvistavat mallin kelvollisuutta. Kohdissa a) ... e) esitetyt ehdot on otettu huomioon tässä simulointimallissa.

3.5.3

Pysähtyminen

Mallissa esiintyy kolmenlaista pysähtymistä: (1) pysähtyminen risteyksen seis-viivalle, (2) pysähtyminen toisen pysähtyneen ajoneuvon perään ja (3) pysähtyminen konfliktitilanteen vuoksi.

Simulointimallia rakennettaessa aluksi tuntui, että edeltäjän seuraamismallilla voitaisiin kuvata myös ajoneuvon pysähtymistä. Risteyksen seis-viivalle pysähtyvän ajoneuvon kuvaamiseen ajateltiin menetelmää sovellettavan siten, että seis-viivalle asetettaisiin kuviteltu pistemäinen ajoneuvo, jolloin ensimmäisellä risteystä lähestyvällä ajoneuvolla olisi edeltäjäajoneuvonaan pysähtynyt kuviteltu ajoneuvo. Kuitenkin jo mallinrakennuksen alkuvaiheessa tuli ilmeiseksi, että sellainen malli ei kuvaa todellista pysähtymistapahtumaa. Pelkkien edeltäjän seuraamismallin yhtälöiden simuloinnin kokeilu HP-tietokoneella osoitti tämän. Havaintoni mukaan kirjallisuudessa ainoastaan Gerlough ja Wagner /11/ mainitsevat tästä edeltäjän seuraamismallin soveltumattomuudesta pysähtymisen simulointiin. Vaikeus johtuu siitä, että ajoneuvon lopullinen pysähtymispaikka ei ole tavoittepysähtymispaikka, vaan joko sitä ennen tai sen jälkeen.

Artikkelin /11/ mukaan Ohio State University:ssä tehty empiirinen työ on osoittanut, että vakiojarrutusmallin käyttö on aivan realistista. Työn mukaan ajajat pyrkivät pysähtymään käyttäen likimäärin vakiohidastuvuutta jarruttaessaan. Työssä on lisäksi määritetty hidastuvuuden keskiarvo ja varianssi empiirisestä aineistosta.

Tässä simulointimallissa käytetyn pysähtymismallin tarvitsemat halutun jarrutushidastuvuuden parametrit luetaan syöttötietoina, ja halutun jarrutushidastuvuuden arvo määritetään satunnaisprosessilla kullekin ajoneuvolle kohdan 3.4.1 mukaisesti. Kun ajoneuvo ei ole pakotettu käyttämään nopeampaa jarrutusta, se alkaa pysähtyä likimäärin halutun jarrutushidastuvuuden mukaan, minkä tuloksena ajoneuvo pysähtyy satunnaisen pysähtymispaikan virheen tarkkuudella ennalta laskettuun pysähtymispaikkaan, jos pysähtymistä vaativat olosuhteet eivät muutu.

Pysähtymismalli voidaan esittää seuraavana kiihtyvyyden yhtälönä:

$$a(t) = \frac{v(t)^2}{2 \cdot [x(t) - (x_{\text{pys}} + \text{ERR})]} , \quad (3-12)$$

jossa

- $a(t)$ on kiihtyvyys (negat.) hetkellä t ,
- $v(t)$ on nopeus hetkellä t ,
- $x(t)$ on paikka hetkellä t ,
- x_{pys} on tavoitepysähtymispaikka,
- ERR on normaalijakautunut pysähtymispaikan virhe, jonka jakautumalla on tietty keskiarvo ja hajonta.

Kun yhtälöstä 3-12 laskettu kiihtyvyyden arvo $a(t)$ tulee pienemmäksi kuin halutun jarrutushidastuvuuden arvo, ajoneuvo siirtyy noudattamaan pysähtymismallia ja se pysähtyy vakiohidastuvuudella $a(t)$. Tavoitepysähtymispaikan arvo x_{pys} määritetään jonon ensimmäiselle ajoneuvolle jonkin risteyksen geometriseen muotoon liittyvän suureen perusteella (esimerkiksi liikennevalojen sijainti tai konfliktipisteen sijainti). Ajoneuvoille, jotka alkavat pysähtyä toisen ajoneuvon perään, tavoitepysähtymispaikan arvon määrittää pysähtyvän edeltäjäajoneuvon efektiivisen perän paikka.

Kaavana tämä voidaan ilmaista muodossa:

$$x_{\text{pys}}(I_{\text{sa}}) = x_{\text{pys}}(I_{\text{ea}}) - \text{PIT}_{\text{eff}}(I_{\text{ea}}) \quad , \quad (3-13)$$

jossa

$x_{\text{pys}}(I_{\text{sa}})$ on seuraaja-ajoneuvon tavoitepysähtymispaikka ,
 $x_{\text{pys}}(I_{\text{ea}})$ on edeltäjäajoneuvon tavoitepysähtymispaikka ,
 $\text{PIT}_{\text{eff}}(I_{\text{ea}})$ on edeltäjäajoneuvon efektiivinen pituus (vrt. kohta 3.4.1).

3.5.4

Kääntyminen

Niitten ajoneuvojen, joiden on määrä kääntyä risteyksessä joko oikealle tai vasemmalle, on luovuttava noudattamasta ärsyke-reaktio-mallia ja ryhdyttävä noudattamaan jotakin riippumatonta kääntymismallia.

Perusvaatimus on, ettei ajoneuvo kääntyessään saa ylittää tiettyä maksiminopeutta. Maksimikääntymisnopeuksille on esitetty erilaisia laskentaperusteita. Esimerkiksi se riippuu kääntymissäteestä ja vierintäkitkasta yhtälön

$$v_{\text{käänt}} = \sqrt{f g r} \quad (3-14)$$

mukaan, jossa

$v_{\text{käänt}}$ on maksimikääntymisnopeus ,
 f on kitkakerroin ,
 r on kääntymissäde ,
 g on painovoimavakio .

AASHO Policy on Geometric Design ilmoittaa, että 95 prosenttisesti pienehköillä kääntymisnopeuksilla kitkakertoimen arvo olisi 0,3.

Tässä mallissa kullekin ajoneuvolle oletetaan annetuksi maksiminopeudet, joten malli ei laske maksimikääntymisnopeuksiakaan, vaan ne on arvioitava jo syöttötietoja annettaessa.

Ajoneuvon lähestyessä risteystä se on käsiteltävänä luonnollisestiikin joka aika-askel ja jos sen saama nopeus ylittää maksimikäntymisnopeuden, ajoneuvo alkaa jarruttaa siten, että se saavuttaa kääntymisnopeuden juuri saapuessaan kääntymiskohtaan. Ajoneuvon kiihtyvyys lasketaan kaavasta

$$a(t) = \frac{v_{\text{käänt}}^2 - v(t)^2}{2[x_{\text{käänt}} - x(t)]}, \quad (3-15)$$

jossa

- $v_{\text{käänt}}$ on maksimikäntymisnopeus,
- $v(t)$ on ajoneuvon nopeus hetkellä t ,
- $x_{\text{käänt}}$ on kääntymisen alkukohta,
- $x(t)$ on ajoneuvon sijainti hetkellä t .

Heti kun kaavasta 3-15 laskettu kiihtyvyysarvo tulee pienemmäksi kuin haluttu jarrutushidastuvuus, ajoneuvo alkaa hidastaa maksimikäntymisnopeuteensa asti hidastuvuuden arvolla $a(t)$, ellei sen liikettä estä jokin vieläkin rajoittavampi tilanne.

Koko kääntyminen tapahtuu maksimikäntymisnopeuteen pyrkien, minkä jälkeen ajoneuvo siirtyy toiselle ajouralle ja sen käyttäytymistä simuloidaan ärsyke-reaktio-malleilla (edeltäjän seuraaminen tai vapaa eteneminen).

Kaikki ajoneuvot eivät voi suorittaa kääntymistä maksimikäntymisnopeudella, vaan esimerkiksi jonossa ollessaan edeltäjäajoneuvot vaikuttavat niiden etenemiseen. Erikoisesti vasemmalle kääntyvät ajoneuvot joutuvat helposti konflikteihin, koska niiden on ylitettävä vastaantulevan liikenteen ajoradat. Simulointimallissa vasemmalle kääntyville ajoneuvoille ei ole omaa päätöksentekomalliaan, vaan niiden kuten kaikkien muidenkin ajoneuvojen etenemistä ohjataan konfliktikäyttäytymismallilla, jota käsitellään kohdassa 3.6.3.

3.5.5. Jonon purkaantuminen

Liikenteessä esiintyvien ajoneuvojonojen käyttäytymisen kuvaamiseen on sovellettu monia jonoteorian menetelmiä kinemaattisesta aaltoteoriasta stokastisiin malleihin ja Markovin prosesseihin /29, 34, 35/. Risteyšliikenteessä jonon purkautumisprosessia tarkasteltaessa ilmenee vapaata ja estettyä purkaantumista. Vapaalla purkaantumisella tarkoitetaan tilannetta, jossa ajoneuvo on riippuvainen vain edeltäjäajoneuvostaan. Useiden tutkijoiden mukaan (mm. Larsson ja Lundin /22/) vapaasti jonosta purkaantuvien ajoneuvojen eteneminen risteyksessä tapahtuu tarkkaan ennustettavissa olevalla tavalla.

Koska tässä mallissa on valittu käytettäväksi ajajan ärsyke-reaktiomalleja ajoneuvojen etenemisen kuvaamiseen, jonon purkaantumisen kuvaaminen on helposti toteutettavissa, kun ajoneuvojen välimatkat ovat kuljettajan ja ajoneuvon ominaisuuksista riippuvina eksaktisti laskettavissa. Vaikka tällä tavoin siirtämällä "black boxeja" taaksepäin selitettävässä suunnassa ohjelman monimutkaisuus kasvaa melkoisesti verrattuna simulointitapaan, jossa ajoneuvojen eteneminen mallissa tapahtuisi yksistään annettujen välimatkajakautumien perusteella, mallista saatavan tiedon yksityiskohtaisuus (mm. etäisyydet edeltäjäajoneuvoon) palkitsee vaivan.

On kaksi "normaalia" syytä, jonka vuoksi jonon purkaantuminen tulee estetyksi. Ensiksi: muun suuntainen liikennevirta estää ajoneuvoa etenemästä (varsinkin kääntyvät ajoneuvot joutuvat usein estetyiksi, jollei kääntyvälle liikenteelle ole omaa liikennevalo-ohjausta). Toiseksi: ajoneuvon edessä olevat samaan suuntaan etenevät ajoneuvot ovat pysähtyneet. Estetyn purkaantumisen käsittelystä on tarkempi kuvaus kohdassa 3.6.3.

Vapaasti purkaantuvan jonon malliksi soveltuu Gerloughin /11/ mukaan hyvin virtausteorian tasapainoyhtälö

$$q = a_0 \cdot K \cdot \log(K_j/K) \quad , \quad (3-16)$$

jossa

q on virtaus

a_0 on karakteristinen nopeus

K on ajoneuvotiheys

K_j on pysähtyneen jonon tiheys

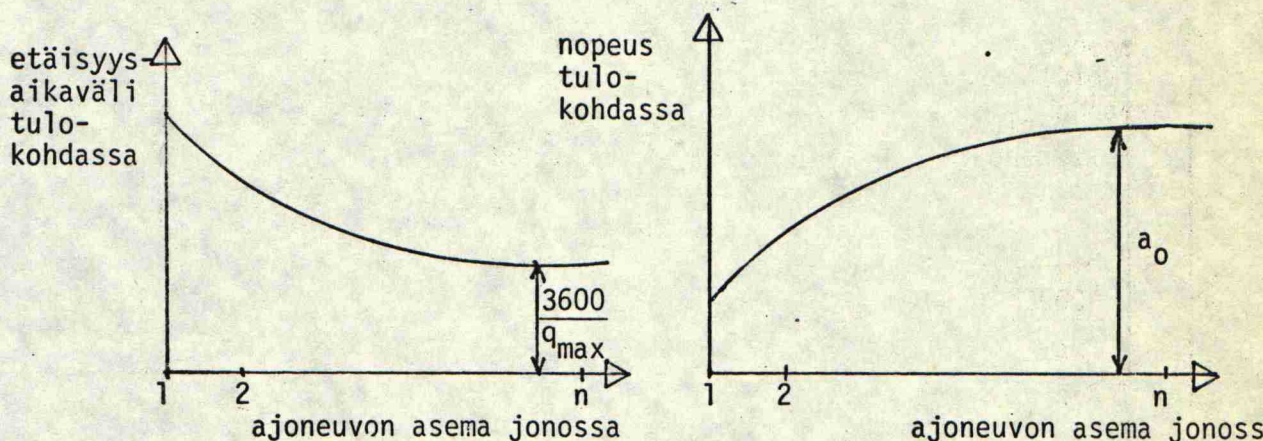
Vaikka tätä yhtälöä on sovellettu onnistuneesti maantieliikenteen ja tunneliliikenteen virtojen tutkimisessa, on se /11/:n mukaan tietyissä risteystilanteissa myös erinomaisen sopiva. Tällainen tilanne syntyy vihreän liikennevalon sytyttyä, jolloin liikennevirta on maksimissaan. Differentioimalla ja asettamalla

$$\frac{dq}{dK} = 0$$

havaitaan, että kun q lähestyy q_{\max} :a, K_j/K lähestyy e:tä.

Valitsemalla K_j :lle joitakin tyypillisiä arvoja, q_{\max} ja a_0 voidaan laskea. (q_{\max} on analoginen kyllästymisvirralle - itse asiassa määritelmästä riippuen nämä kaksi arvoa ovat sama asia suoraan etenevälle ajoneuvovirralle).

Eräs Gerloughin ja Wagnerin /11/ havaitsema tärkeä seikka on se, että yksilöajoneuvon käyttäytymismalli (vapaa eteneminen), vuorovaikutuksellinen etäisyysmalli (edeltäjän seuraaminen) ja liikennevirran tasapainoyhtälö 3-16 ovat yhteensopivia.



Kuva 3 - 8. Jonon purkaantuminen.

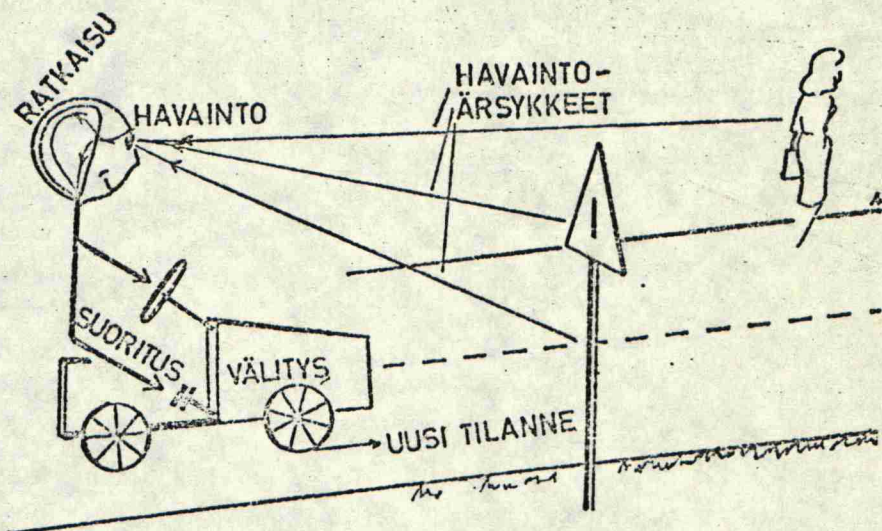
Karakteristinen nopeus a_0 yhtälössä 3-16 on sama, jota käytetään edeltäjän seuraamismallin yhtälössä 3-9 silloin, kun mallin 3-9 ajajan herkkyystekijä ja ajoneuvon nopeustekijä on yhdistetty. Täten olisi mahdollista laskea aina yhtälön 3-9 karakteristinen nopeus, kun risteyksen sisääntulokohdan ohittanut maksimaalinen liikennevirtaus on mitattu. Jos sovelletaan vuorovaikutuksellista etäisyysmallia jonossa oleviin ajoneuvoihin ja vapaan etenemisen mallia jonon ensimmäiseen ajoneuvoon, saadaan artikkelissa /11/ esitetyn kuvan 3-8 mukaisia riippuvuuksia ajoneuvojen välisille etäisyyksille ja nopeuksille.

3.6 Ajajan käyttäytymismallit

3.6.1 Yleistä havaitsemisesta, päätöksenteosta ja reaktioajasta

Ajoneuvojen etenemistä ohjaavissa deterministisissä käyttäytymismalleissa (kohta 3.5) oletetaan kuljettajan havaintotoiminnan olevan täydellistä ts. hän havaitsee kaikki turvallisen etenemisen edellyttämät seikat. Koska näitä malleja noudattavien ajoneuvojen käyttäytymiseen katsotaan vain edeltäjäajoneuvon vaikuttavan, tehty oletus on hyväksyttävissä.

Todellisuudessaan ajoneuvon kuljettajan ajotoiminta sisältää



Kuva 3 - 9 . Autonkuljettajan ajotoiminnan kolme vaihetta: havainto, ratkaisu ja suoritukset.

kolme toisiaan seuraavaa ja toisistaan riippuvaa vaihetta: havainnonteko, päätöksenteko (ratkaisu) ja suoritukset. Uusi tilanne merkitsee uusia havaintoja, ratkaisuja ja säätöliikettä (kuva 3- 9, /16/). Simuloitaessa tilanteita, joissa ajoneuvojen etenemistä ohjaavat stokastiset käyttäytymismallit, joudutaan ajotoiminnan eri vaiheet käsittelemään erillisinä.

Kuljettajan toimintojen määrä ja laatu vaihtelevat erittäin paljon eri liikennetilanteissa ja ajotehtävissä. Vilkaassa liikenteessä havaintoärsykykeitä voi olla niin paljon tai ne esiintyvät niin nopeassa tahdissa, että kuljettajan kapasiteetin yläraja ylittyy. Taulukossa 3-2 /16/ on luettelo ajotapahtumista ja niiden esiintymistiheyksistä vilkaassa liikenteessä.

Taulukko 3 - 2. Ajotapahtumat ja niiden esiintymistiheydet vilkaassa liikenteessä.

Tapahtuma	Lukum.	Aika
liikennetapahtumia	5	1 sekunnissa
kuljettajan havaintoja	2	1 sekunnissa
-"- ratkaisuja	40	1 minuutissa
-"- suorituksia	30	1 minuutissa
-"- virheitä	1	2 minuutissa
vaarallisia tilanteita	1	2 tunnissa
lähes-onnettomuuksia	1	1 kuukaudessa

perusteet: keskinopeus 60 km/h

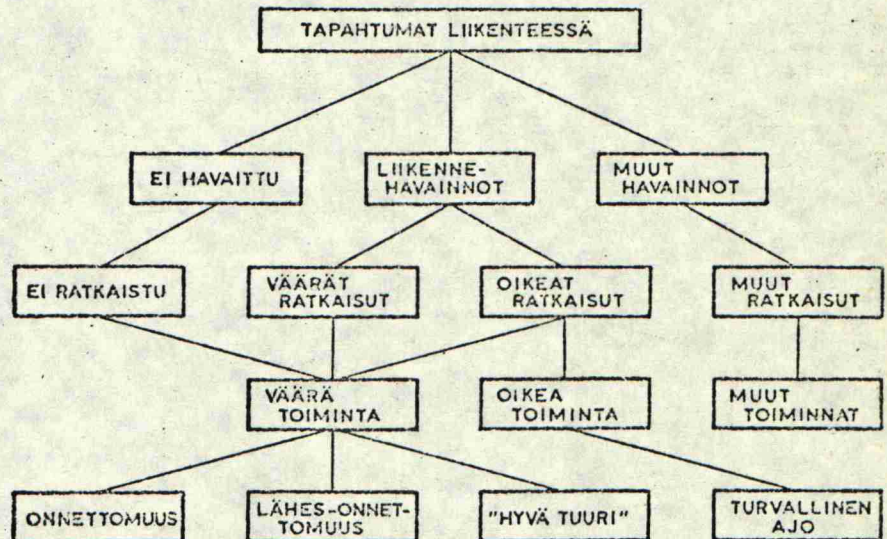
ajokilometrimäärä 16000 km vuodessa

Liikennetapahtumalla ymmärretään tapahtumaa, joka sattuu kuljettajan välittömässä läheisyydessä. Se voi vaikuttaa tai olla vaikuttamatta hänen käyttäytymiseensä. Osa tapahtumista jää havaitsematta, sillä ihminen pystyy tekemään erillisiä havaintoja enintään kaksi sekunnissa. Tämän vuoksi simuloinnin aika-askel ei saisi olla paljoa 0,5 sekuntia pitempi (vrt. kohta 3.2). Osa havainnoista aiheuttaa ratkaisunteen ja osa säästösuorituksia. Häkkinen/16/mukaan kuljettajan on todettu tekevän virheitä keskimäärin

joka toinen minuutti. Virhe merkitsee tässä sitä, että kuljettaja ei ole reagoinut tilanteeseen adekvaatilla tavalla. Taulukko 3-2 osoittaa, kuinka ajotoiminta koostuu pienistä osatoiminnoista, jotka toisinaan ovat virheellisiä. Näistä virheistä on silloin tällöin seurauksena häiriö liikenteen kulussa jota tässä nimitetään konfliktiksi. Virheellinen toiminta voi aiheutua kolmella eri tasolla tehdystä virheestä (kuva 3-10).

Kyseistä liikennetapahtumaa ei ensinkään havaittu tai havainto on virheellinen. Tilanne havaitaan, mutta tehdään väärä ratkaisu (esimerkiksi luullaan, että risteykseen samanaikaisesti tuleva toinen ajoneuvo pysähtyy) tai päätös on oikea, mutta suoritus on virheellinen (päätetään, että nopeutta on vähennettävä, mutta liukkaalla kelillä toimitaan väärin).

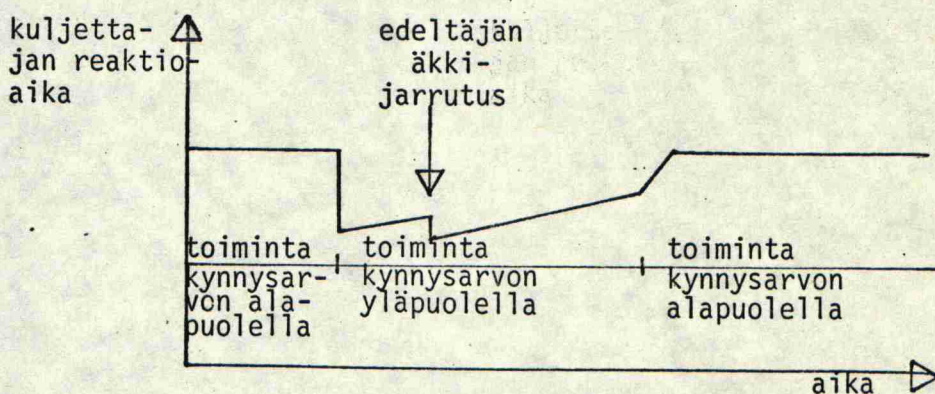
Simulointimallissa puutteellinen havaintotoiminta ja väärien ratkaisujen mahdollisuus ovat mukana stokastisina osamalleina, mutta ajoneuvojen käyttäytymisen oletetaan aina olevan havainto-



Kuva 3 - 10. Kuljettajan toiminnan eri vaiheet ja virhemahdollisuudet.

jen ja päätösten mukainen. Mallissa suoritetaan yksinkertaistettu havaitseminen ja päätöksenteko siten, että kun päätös on tehty havaitsemisen jälkeen, sitä ei enää muuteta. Esimerkiksi ajajan päätettyä jollakin hetkellä ajaa keltaisista valoista pidetään tämä päätös automaattisesti voimassa myös seuraavana ajanhetkenä. Epätäydellistä havaitsemista oletetaan esiintyvän vain ajoneuvon ajoreittiä risteävän liikenteen suhteen (kohta 3.6.3). Sen sijaan havaitsemisen oletetaan olevan täydellistä muulloin (esimerkiksi liikennevaloja lähestyttäessä ja edeltäjäajoneuvoa seurattaessa).

Näkemisen, havaitsemisen ja tiedostamisen kuluttamat aikaviiveet kussakin kuljettajan emotionaalisessa tilassa yleensä yhdistetään kuljettajan reaktioajaksi ulkoisen ärsykkeen suhteen. Vaikka inhimillisen käyttäytymisen tästä puolesta on kirjoitettu paljon, ei siitä ole saatavissa eksaktia kvantitatiivista tietoa. Forbes /9/, Fox ja Lehman /10/ ja useat muut ovat tutkineet kuljettajan reaktioaikaan vaikuttavia useita tekijöitä. Reaktioaika vaihtelee kuljettajasta kuljettajaan ja jopa saman kuljettajan kohdalla tilanteesta toiseen. Tyypillinen reaktioajan vaihtelu on /10/:n mukaan kuvan 3-11 mukainen. Esimerkiksi ärsyke-reaktio-mallia tarkasteltaessa reaktioaika lyhenee huomattavasti kuljettajan noudattaessa edeltäjän seuraamismallia verrattuna nopeuden muutoksen havaitsemisen kynnyksen alapuolella noudatettavan vapaan etenemismallin aikaiseen reaktioaikaan. Jälleen kuljettajan reaktioaika kasvaa ajan kuluessa, kunnes se saavuttaa maksimiarvonsa, joka riippuu ajoneuvon noudattamasta etenemismallista. Aina kun odottama-



Kuva 3 - 11. Kuljettajan reaktioaika.

tonta tapahtuu, kuten edeltäjäajoneuvon nopea jarrutus, reaktioaika lyhenee.

Reaktioajan arvoiksi on raportoitu 0,29...1,5 sekuntin aikoja, onpa joissakin tapauksissa päädytty nolla-arvoihinkin. Kaupunkiliikenteessä reaktioajan arvoksi on useimmin päädytty 1 sekuntiin. Vaikka todellisuudessa reaktioaika vaihtelee, on kaikissa referoiduissa artikkeleissa käytetty vakioarvoista reaktioaikaa, kuten käsiteltävässäkin työssä, koska vaihtelevan mittaisen reaktioajan huomioonottaminen etenemisyhtälöiden viiveenä olisi hyvin hankalaa.

3.6.2

Keltaista liikennevaloa lähestyminen

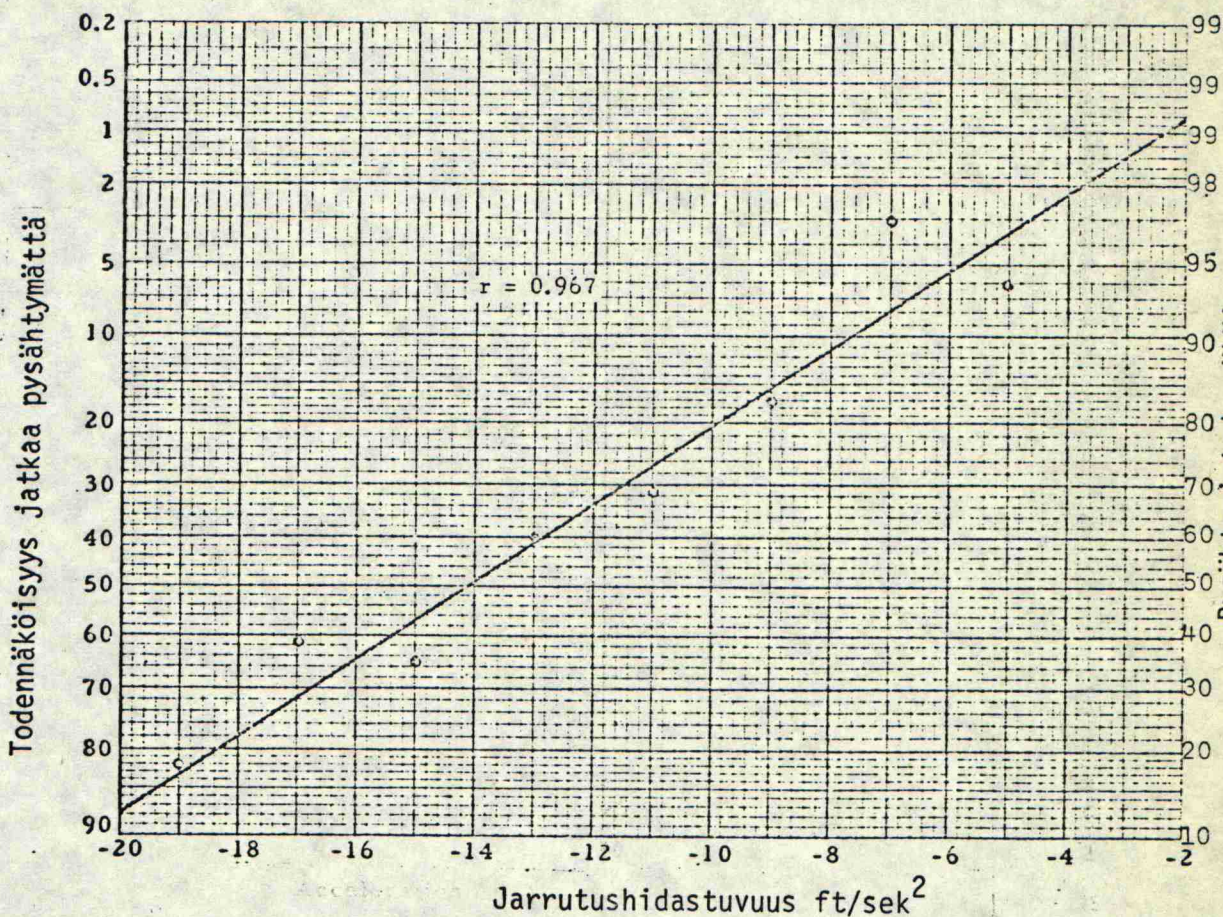
Mallissa erotetaan vihreän liikennevalon ja punaisen liikennevalon jälkeiset keltaiset valomerkit toisistaan. Punaisen valon jälkeisessä keltaisessa valossa ensimmäisenä jonossa oleva ajoneuvo voi päättää lähteä liikkeelle jo keltaisen valon palauksessa, jos sen reitillä ei ole konfliktin uhkaa. Päätös tehdään satunnaisprosessilla.

Vihreän liikennevalon jälkeistä keltaista valoa lähestyvän ajoneuvon käyttäytymisestä on kirjallisuudessa tutkimustuloksia ja empiirisiä dataa. Näiden datajen perusteella on kehitettävissä todennäköisyysjakautumia, joiden mukaan ajajan voidaan olettaa tekevän päätöksensä joko pysähtyä tai jatkaa ajoansa keltaista valoa kohti.

Tässä mallissa käytetty pysähtymistä koskevan päätöksenteon todennäköisyysjakautuma perustuu Gerloughin ja Wagnerin [11]:ssa esittämän probit-menetelmän mukaan laskettuun neljän englantilaisen tutkijan esittämiin datoihin (mm. Olson ja Rothery [32]). Probit-menetelmässä epälineaarinen logaritmi-normaali-funktio, jossa argumentin logaritmi on normaalijakautunut suure, transformoidaan yksinkertaiseen lineaariseen muotoon. [11]:n mukaan todennäköisyysjakautuman laskemisessa käytetään pysähtymistodennäköisyyttä joko keltaisen valon syttymishetkellä risteyksestä mi-

Taulukko 3 - 3. Todennäköisyys pysähtyä keltaisissa liikennevaloissa.

jarrutushidastuvuus ft/sek ²	pysähtymistodennä- köisyys
-0 ... -0.99	1.000
-1.00 ... -1.99	0.994
-2.00 ... -2.99	0.989
-3.00 ... -3.99	0.982
-4.00 ... -4.99	0.972
-5.00 ... -5.99	0.956
-6.00 ... -6.99	0.935
-7.00 ... -7.99	0.905
-8.00 ... -8.99	0.867
-9.00 ... -9.99	0.820
-10.00 ... -10.99	0.762
-11.00 ... -11.99	0.700
-12.00 ... -12.99	0.624
-13.00 ... -13.99	0.548
-14.00 ... -14.99	0.468
-15.00 ... -15.99	0.390
-16.00 ... -16.99	0.318
-17.00 ... -17.99	0.250
-18.00 ... -18.99	0.190
-19.00 ... -19.99	0.140



Kuva 3 - 12. Todennäköisyys pysähtyä keltaisissa liikennevaloissa.

tatun matkaetäisyyden tai ajoajan funktiona. Ajoneuvon pysähtymiseksi seis-viivalle tarvittavien hidastuvuuksien laskemisessa reaktioajaksi oletetaan 1 sekunti. Gerloughin ja Wagnerin mukaan datat useista eri risteyksistä ja ajoneuvojen käyttäessä eri nopeuksia noudattavat tiettyä jakautumaa. Kuvassa 3-12 on piirretty pysähtymistodennäköisyys ajoneuvon pysäyttämiseksi vaadittavan hidastuvuuden funktiona. Keltaisen liikennevalon pysähtymisen todennäköisyyden taulukon (taulukko 3-3) arvoja, jotka ovat kuvan 3-12 mukaisia, käytetään mallissa.

Mallissa on oletettu liikennevalojen havaitseminen täydelliseksi, koska ei ole ollut saatavissa havaitsemisen todennäköisyysjakautumia. Toisaalta sellaiset tilanteet, joissa kuljettaja ei todellisuudessa ole havainnut liikennevaloja, voidaan mallissa ajatella käsiteltävän stokastisen päätöksentekoprosessin vuoksi väärinä ratkaisuina, jotka johtavat väärään toimintaan. Koska väärä toiminta saattaa johtaa konfliktitilanteeseen, tulee liikennevalojen epätäydellinen havaitseminen impliittisesti käsitellyiksi mallin konfliktitilanteissa.

3.6.3

Päätöksenteko toista ajouraa ylitettäessä ja konfliktitilanteissa

Kun ajoneuvo kulkee risteyksen läpi, se joutuu ylittämään risteäviä ajouria, jolloin sen ja risteävän Tiikenteen välisellä vuorovaikutuksella on keskeinen merkitys. Tähän vuorovaikutukseen sisältyy kuljettajan päätöksentekoa ajaessaan ajoneuvoa risteyksessä. Risteävien ajoneuvojen nopeuksien ja etäisyyksien arviointi sekä risteävässä liikennevirrassa välin hyväksymistapahtuma ovat tällaisen liikennevirtavaikutuksen kriittisimpiä аспектеja. Kidd ja Laughery raportoivat /20/:ssa ajajan käyttäytymisen simulointimallista, jossa he tutkivat eräiden sopivien parametrien vaikutusta ajajan suorituskykyyn ja konfliktien määrään dynaamisena säätötehtävänä. Heidän mukaansa tärkeimmiksi parametreiksi osoittautuivat:

a) miten kauaskantoisia päätöksiä ajaja tekee,

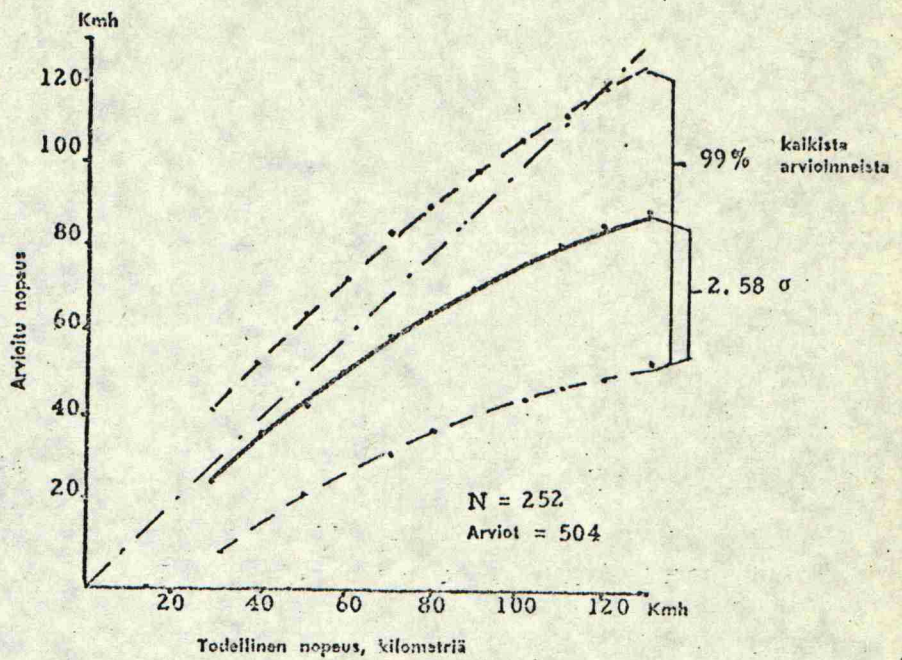
- b) kunkin päätöksentekotilanteen vaatima aika (reaktioaika),
- c) nopeuden muutoksen havaitsemisen kynnyks,
- d) ajajan ajoneuvon nopeus ja
- e) havaintokentässä olevan näköesteen sijainti.

Konfliktipisteissä (, jotka kohdan 3.3.1 määritelmän mukaan ovat kahden ajouran kohtauspisteitä) ajoneuvojen etenemistä ohjaavassa päätöksentekomallissa on otettu huomioon parametrit (a)...(e). Tällaisen kahden ajoneuvon välinen vuorovaikutustilanne on vaikeasti kuvattavissa analyttisin yhtälöin (tilanne muistuttaa peliteorian mukaista kahden pelaajan peliä, jota mahdollisesti voitaisiin käyttää hyväksi), eikä edes ole ollut löydettävissä raportteja siitä, että kyseistä vuorovaikutustilannetta olisi simuloitu. Kaikki referoidut simulointimallit on rakennettu siten, että ei syntyisi konfliktitilanteita muiden ajoneuvojen kanssa.

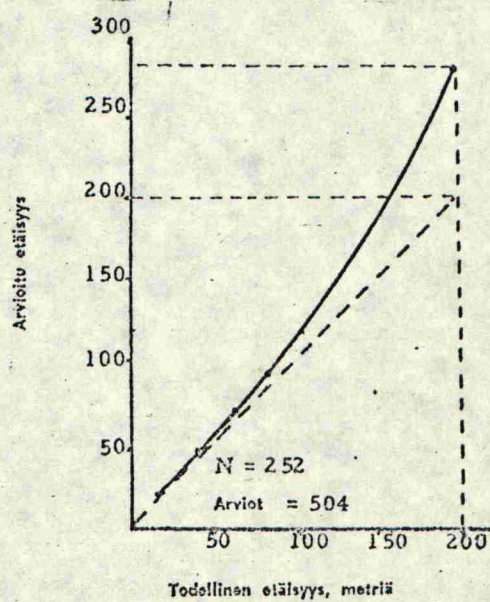
Liikenteessä jatkuvasti suoritettavista nopeuksien ja etäisyyksien arvioinneista Häkkinen on tehnyt tutkimuksen /17/ Suomessa. Sen mukaan nopeuksia aliarvioidaan ja virhe on sitä suurempi, mitä suuremmasta nopeudesta on kysymys (kuva 3-13); vastaavasti etäisyydet arvioitiin liian pitkiksi. Tutkimuksen mukaan virheet nopeuden arvioinnissa ovat riippumattomia arvioinnin kohteesta. Riippuvuus todellisen nopeuden ja koetun nopeuden välillä ei ole lineaarinen, vaan logaritminen. Arvioinnit noudattavat inhimillistä havaitsemista kuvaavaa Weberin lakia jonka mukaisesti myös arviointien varianssien ja keskiarvon suhde eri nopeuksilla pysyy likimain vakiona.

Päätöksentekomallissa ajoneuvon omasta ja toisen ajoneuvon nopeudesta ja etäisyydestä tekemä arviointi tapahtuu kuvien 3-13 ja 3-14 mukaisten datojen avulla.

Useita tutkimuksia /37,39/ on tehty kuljettajan ominaisuudesta, jonka mukaan hän hyväksyy risteävässä liikenteessä olevan aikavälin (lag) tai matkavälin (gap). Näissä tutkimuksissa on kehi-



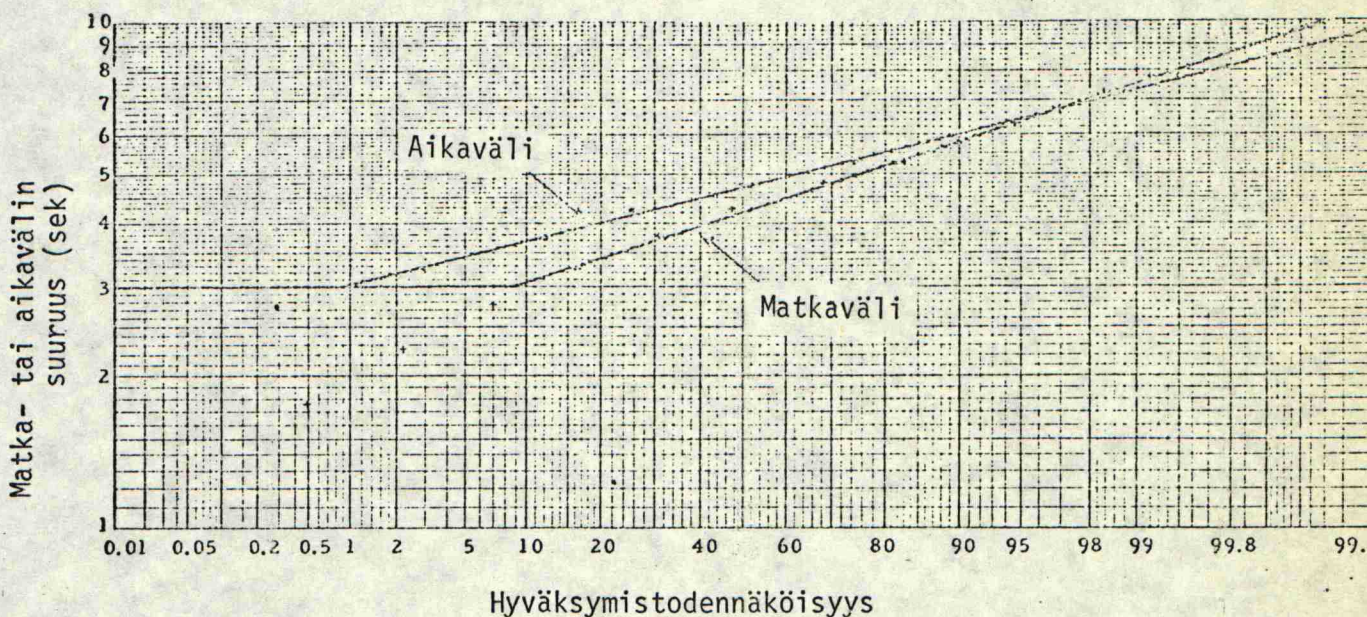
Kuva 3 - 13. Vastaan tulevan auton nopeuden arviointi. Arviointien keskiarvo ja hajonta.



Kuva 3 - 14. Vastaan tulevan auton etäisyyden arviointi. Arviointien keskiarvo.

tetty välin hyväksymisen jakautuma kokeellisten tutkimusten pohjalta. Useilla teoreettisilla todennäköisyysfunktioilla on pyritty kuvaamaan välinhyväksymisilmiötä - negatiivisella eksponettijakautumalla, Erlandin ja logaritmi-normaalilla funktioilla. Logaritmi-normaalinen funktio soveltuu parhaiten mm./36/:n mukaan.

Tässä työssä on käytetty /11/:ssa esitettyä logaritmi-normaalista jakautumaa aikavälin ja matkavälin hyväksymisjakautumina, jotka on esitetty kuvassa 3-15 ja taulukossa 3-4.



Kuva 3 - 15. Aikavälin ja matkavälin hyväksymisjakautumat

Koska aikaväli- ja matkavälijakautumat ovat melkein samantlaisia, mallissa käytetään vain aikavälijakautumaa.

Koska valtaosa onnettomuuksista ja konflikteista sattuu epätodellisen havaintotoiminnan seurauksena, kuvataan havaintotoiminta tässä päätöksentekomallissa stokastisena osamallina. Ajoneuvon todennäköisyys havaita risteävää ajouraa pitkin konfliktipistettä kohti lähenevä ajoneuvo riippuu kummankin ajoneuvon etäisyydestä konfliktipisteeseen ja siitä kulmasta, josta havainto

Taulukko 3 - 4. Aikavälin ja matkavälin hyväksymisen todennäköisyystaulukko.

aika- tai matkaväli	aikavälin hyväk- symisen todennä- köisyys	matkavälin hyväk- symisen todennä- köisyys
0 - 0.5	0	0
0.5 - 1.0	0	0
1.0 - 1.5	0	0
1.5 - 2.0	0	0
2.0 - 2.5	0	0
2.5 - 3.0	0	0
3.0 - 3.5	0.030	0.150
3.5 - 4.0	0.124	0.320
4.0 - 4.5	0.300	0.590
4.5 - 5.0	0.530	0.690
5.0 - 5.5	0.730	0.820
5.5 - 6.0	0.860	0.900
6.0 - 6.5	0.940	0.950
6.5 - 7.0	0.970	0.970
7.0 - 7.5	0.990	0.986
7.5 - 8.0	0.996	0.993
8.0 - 8.5	0.999	0.997
8.5 - 9.0	1.000	0.998
9.0 - 9.5	1.000	0.999
9.5 - 10.0	1.000	1.000

tehdään. Havaitsemisen todennäköisyysfunktion oletetaan riippuvan eksponentiaalisesti ajoneuvon suhteellisista etäisyyksistä ja cos-funktion mukaan havainnontekokulman puolikkaasta. Tällöin havaitsemisen todennäköisyys kasvaa ajoneuvojen lähestyessä konfliktipistettä, jossa se saavuttaa arvon 1, kun ajoneuvojen nokat kolahtavat toisiinsa. Lisäksi havaitsemista haittaavana tekijänä voidaan viereisellä ajouralla näköesteinä olevat ajoneuvot ottaa huomioon generoimalla ne stokastisiksi näköesteiksi. Mikäli ajoneuvot ovat tiettyä rajaa kauempana konfliktipisteestä,

niiden havaitsemisen todennäköisyysfunktio on nolla-arvoinen. Havaitsemisen todennäköisyysfunktio on kuvattavissa muodollisesti kaavalla

$$P\{A \text{ havaitsee } B:n\} =$$

$$e^{-x_A/a_x} \cdot e^{-x_B/b_x} \cdot \cos\left(\frac{\alpha_{hav}}{2}\right) \cdot P\{\text{ei näköestettä}\}, \quad (3-14)$$

jossa

x_A/a_x on A:n suhteellinen etäisyys konfliktipisteestä

x_B/b_x on B:n suhteellinen etäisyys konfliktipisteestä

α_{hav} on kulma jossa A havaitsee B:n
A:n kulkusuunnan suhteen.

Kuvassa 3-15 todennäköisyysfunktio on esitetty graafisesti ajoneuvojen etäisyyksien funktiona eräillä havaintokulman ja näköesteen todennäköisyyksien arvoilla. Koska risteysliikenteen havainnonteosta ei ollut saatavissa minkäänlaista tutkimusaineistoa, tuntuu "hihasta vedetyn" yhtälön 3-14 validisointi tässä työssä irrelevantilta.

Kohdan 3.6.3 alussa esitetyistä parametreista tärkeimmäksi osoittautunut päätöksenteon kauaskantoisuus on tässä mallissa mukana argumenttina siten, että sillä ilmoitetaan ajoneuvon ajoreititään kerralla tutkimien konfliktipisteiden lukumäärä.

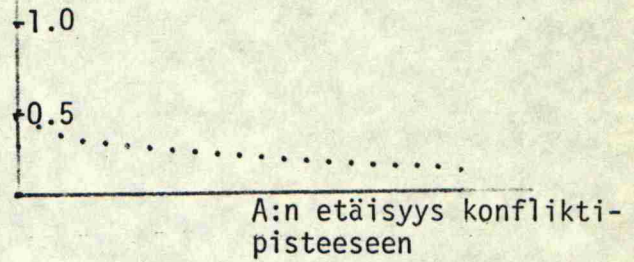
Mallissa etuajo-oikeus otetaan huomioon aikavälin hyväksymisen testauksen yhteydessä siten, että etuajo-oikeutettu ajoneuvo alkaa jarruttaa uhkaavan konfliktin vuoksi mahdollisimman myöhään välttämällä kuitenkin onnettomuudeksi luettavan konfliktin. Väistämisvelvollinen ajoneuvo sen sijaan alkaa jarruttaa heti havaittuaan etuajo-oikeutetun ajoneuvon lähestymisen, kuitenkin jos jarruhidastuvuus on haluttua jarrutushidastuvuutta hitaampi, vasta tämän rajahidastuvuuden alapuolella. Mallissa etuajo-oikeus määräytyy yksinomaan oikealta lähestymisen perusteella. Liikenteeseen liittyvistä tottumuksista ja luontaisista hahmolaeista aiheutuu, että eräissä olosuhteissa tietyt ajosuunnat käsi-

$P(A \text{ havaitsee } B:n)$

$v_A = v$ ja $v_B = 0$

alkuetäisyydet:

$x_A = 20$ ja $x_B = 30$

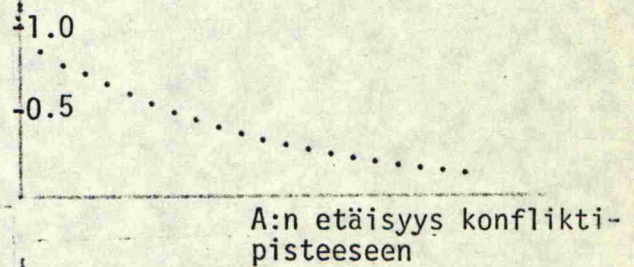


$P(A \text{ havaitsee } B:n)$

$v_A = v$ ja $v_B = 1,5*v$

alkuetäisyydet:

$x_A = 20$ ja $x_B = 30$

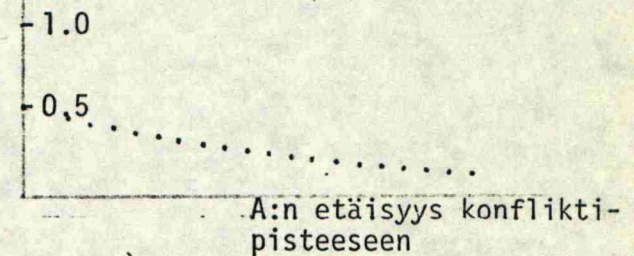


$P(A \text{ havaitsee } B:n)$

$v_A = v$ ja $v_B = 0,5*v$

alkuetäisyydet:

$x_A = 20$ ja $x_B = 10$

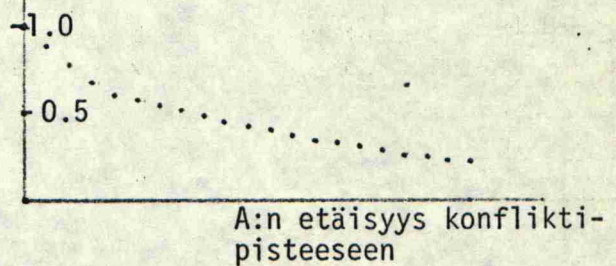


$P(A \text{ havaitsee } B:n)$

$v_A = v$ ja $v_B = v$

alkuetäisyydet:

$x_A = 20$ ja $x_B = 30$



Kuva 3 - 16. Havaitsemisen todennäköisyysfunktion graafinen esitys eräillä parametrien arvoilla.

tetään luontaisesti etuajo-oikeutetuiksi. Näissä tapauksissa voidaan puhua "psykologisesta etuajo-oikeudesta", josta muutamia esimerkkejä /31/:

Samanarvoisten teiden tai katujen risteyksissä n. 90 % kuljettajista vähentää nopeutta ja tarkkailee oikealta tulevaa liikennettä, kun tiet ovat yhtä leveitä, samalla tavalla päällystettyjä ja liikennetiheys suunnilleen sama. Kapean ja leveän kadun risteyksessä vain 50 % leveällä kadulla ajavista vähentää nopeutta ja tarkkailee oikealle. Tapauksissa, jolloin kapeampi katu on etuajo-oikeutettu, 90 % sillä ajavista kuljettajista ei uskalla käyttää etuajo-oikeuttaan, vaan epäilee, hidastaa ajoaan ja jää odottamaan leveällä kadulla ajavia. Etuajo-oikeutetuksi kaduksi koetaan edelleen esimerkiksi katu, jolla on raitiovaunuliikennettä, tai T - risteyksen suora osa. Eri ajoneuvoluokkien välinen psykologinen etuajo-oikeus on myös olemassa ajoneuvojen koon mukaan (bussi, henkilöauto) tai niiden liikkumistavan mukaan (auto, raitiovaunu).

Sovellettaessa simulointimallia varten kerättyä havaintoaineistoa myös psykologinen etuajo-oikeus on helposti otettavissa mukaan.

3.7

Jalankulkijat ja ympäristötekijät

Jalankulkijat on otettu huomioon mallissa kollektiivisesti siten, että konfliktipisteissä niillä on esiintymistodennäköisyysjakautuma, joka puolestaan riippuu vastaavan suojatien liikennevalojen tilasta ja kestosta. Jos risteys ei ole liikennevaloin ohjattu, todennäköisyysjakautuma muuttuu syöttötietoina luetuksi todennäköisyysluvuksi. Ajoneuvot voivat joutua konfliktitilanteisiin jalankulkijoiden kanssa aivan vastaavalla tavalla kuin toisten ajoneuvojen kanssa.

Risteyksen ohjaussysteemi (liikennevalot) ja liikenteen etenemisen kannalta muut ympäristötekijät, kuten sää, valaistus ja näkyvyys vaikuttavat huomattavasti liikenteen käyttäytymisömi-

naisuuksiin. Vain ensimmäinen tekijä (liikennevalot) voidaan ottaa mallissa huomioon. Tästä lähemmin kohdassa 4.2.8. Muista ympäristötekijöistä ei ole saatavissa sitä välttämättömyyttä informaatiota, joka mahdollistaisi sen, että tällaiset vaikutukset voitaisiin ottaa huomioon simulointimallissa. Tämän vuoksi ympäristöolosuhteet on oletettava olevan ideaaliset sikäli, kun tarkastetaan niiden vaikutusta liikenteeseen.

3.8

Ajoneuvojen generointi ja poistaminen risteyssysteemistä

Koska vain harvoin on mahdollista käyttää mitattuja ajoneuvovälijakautumia simulointimalleissa, teoreettiset jakautumat ovat varteenotettava vaihtoehto. Kirjallisuudessa käsitellyissä risteysmalleissa on usein saapuvien ajoneuvojen generoinnin tilanteita idealisoitu niin, että on voitu käyttää Poissonin ja jopa Bernoullin jakautumia. Valitettavasti tällaiset oletukset eivät vastaa todellisuutta kaupunkiliikenteessä, jossa risteyskseen saapumisprosessilla on usein voimakkaastikin riippuva piirre, jolloin generointiehdot ovat Markovin prosessin mukaisia. Kirjallisuudessa yleisimmin esiintyneet ajoneuvovälijakautumat ovat negatiivinen eksponenttijakautuma, siirretty eksponenttijakautuma, hyper-eksponenttijakautuma, modifioitu binomijakautuma, Erlang-jakautuma ja hyper-Erlang-jakautuma.

Tässä työssä on käytetty ajoneuvojen generoinnissa risteys-systeeminrajalle kussakin generointipisteessä hyper-eksponenttijakautumaa, joka koostuu kahdenlaisten ajoneuvojen generointiaikavälijakautumista. Liikennevirta voidaan ajatella koostuvan "vapaasti etenevien" ja "rajoitetusti etenevien" ajoneuvojen luokista. Yhdistetty jakautuma on muotoa:

$$P\{\text{aikaväli} \geq t\} = \alpha \cdot \exp\left(-\frac{t-\lambda}{T_1-\lambda}\right) + (1-\alpha)\exp\left(-\frac{t-\tau}{T_2-\tau}\right), \quad (3-18)$$

jossa

$P\{\text{aikaväli} \geq t\}$ on todennäköisyys, että aikaväli on suurempi kuin t ,

- α on vapaasti etenevän liikenteen osuus,
- $1 - \alpha$ on rajoitetusti etenevän liikenteen osuus,
- T_1 vapaasti etenevien ajoneuvojen keskimääräinen aikaväli,
- T_2 on rajoitetusti etenevien ajoneuvojen keskimääräinen aikaväli,
- λ on vapaasti etenevien ajoneuvojen pienin sallittu aikaväli,
- τ on rajoitetusti etenevien ajoneuvojen pienin sallittu aikaväli.

Kun keskimääräiset liikennemäärät ja minimiaikavälit on annettu, voidaan muut termit määrittää ja saadaan yksikäsitteinen jakautuma. Gerloughin [11] mukaan jakautumasta 3-18 saadut aikavälit ovat hyvin sopivia kenttätutkimuksista saatujen datojen kanssa.

Kun ajoneuvo on generoitu, sille määritetään sopivat ominaisuudet satunnaisesti kohdassa 3.4.1 esitetyistä jakautumista.

Kun risteysysteemin läpi kulkeneet ajoneuvot ovat ulosmenokaistansa päässä, ne poistetaan systeemistä ja ajoneuvoon liittyvät tilastot viedään "ajoneuvorekisteriin". Samalla kun ajoneuvo poistetaan simulointisysteemistä, on huolehdittava siitä, että mahdollisesti jäljelle jäänyt tavoitenopeutensa alapuolella etenevä seuraaja-ajoneuvo ei ala kiihdyttää vapaan etenemisen mallin mukaan. Tämän vaikutuksen eliminoimiseksi voidaan esimerkiksi joko asettaa kaksi peräkkäistä poistumisrajaa tai pakottaa seuraaja-ajoneuvo vakioliikkeeseen. Tässä mallissa on käytetty jälkimmäistä menetelmää.

3.9

Liikennesuureiden mittauksista mallilla

Kun jokaista mallissa mukana olevaa ajoneuvoa käsitellään simuloinnin kuluessa joka aika-askel, on helposti toteutettavissa tarkastus, jossa määritetään ajoneuvon asema tietyn ennalta-asetetun kontrollipisteen suhteen. Jos halutaan, joukko ajoneuvon operatiivista toimintaa kuvaavia suureita

- on mitattavissa, laskettavissa ja talletettavissa jollekin tietovälineelle simuloinnin jälkeistä tilastollista analyysää varten.

Eräs kehitetyn mallin erittäin tärkeä piirre operatiivisten mittausten kannalta on se, että simulointiohjelma on perioidinen tasa-aika-askel -tyyppiä. Simuloitava reaaliaika on aika-askelen tarkkuudella jatkuvana suurena aina tiedossa. Yhtä tärkeä on, että systeemissä olevien yksittäisten ajoneuvojen ominaisuudet ovat koko ajan selvästi toisistaan erotettavissa.

Sovellettavia liikennesuureiden mittauksia voisivat esimerkiksi olla seuraavat:

1. Ajoneuvojen lukumäärämittaus, jossa ilmaisain ja laskuri voitaisiin sijoittaa risteyksen mielivaltaiseen kohtaan ja laskea ohi kulkeneiden ajoneuvojen lukumäärä.
2. Ajoneuvon olemassaolon mittaus, jossa risteykseen sijoitettu ilmaisain ilmoittaisi, onko ko. paikassa ajoneuvoa vai eikö ole.
3. Ajoneuvon nopeusmittaus, jossa risteykseen sijoitettu nopeusmittari laskee ko. paikan ohittaneiden ajoneuvojen nopeuksista tilastoa.
4. Ajoneuvojen välisen matkaetäisyyden mittaus.
5. Ajoneuvovirran tiheyden mittaus.
6. Ajoneuvojen välisen aikaetäisyyden mittaus.
7. Jonon pituuden mittaus, jossa voitaisiin myös liikkuvan jonon pituudet mitata.

Köhdassa 4.3 selostetussa konstruoidun mallin ohjelmoinnissa on operatiivisista liikennesuureiden mittauksista vain 7. (jonon pituuden mittaus) toteutettu. Ko. toteutuksessakin jonon pituuden mittaus suoritetaan pysähtyneestä jonosta.

4. SIMULOINTIMALLIN OHJELMOINTI

4.1 Simulointikieli

Vaikkakin useimmat liikennesimulointiohjelmat on kirjoitettu probleemaorientoituneilla kielillä, FORTRAN:lla ja ALGOL:lla on olemassa useita erikoissimulointikieliä. GPSS, SIMSCRIPT ja SIMULA ovat kolme tällaista kieltä.

GPSS on jonotustyyppisiin malleihin kehitetty simulointikieli, jossa mallit rakennetaan standardisoiduista lohkoerkoista. Systeemiaktiiviteetteja esittävät nämä lohkot ja systeemin tilan muutokset esitetään transaktioilla, jotka liikkuvat lohkoverkossa simuloinnin aikana.

SIMSCRIPT on yleissimulointikieli, jonka tärkeimmät käsitteet ovat olio, tapahtumarutiini ja joukko. SIMSCRIPT:n ohjelmointitehokkuus on paljon suurempi kuin GPSS:n, tosin se on myös melkoisesti monimutkaisempi kuin GPSS varsinkin SIMSCRIPT 1.5 -version jäykkien määrittely- ja initialisointimuotojen vuoksi.

SIMULA on myös yleistarkoituksiin tehty ohjelmointikieli, jossa SIMSCRIPT:iin verrattuna on voimakas prosessi-käsite, jonka vaikutuksesta myös ohjelman luetettavuus helpottuu. SIMULA:n ohjelmointitehokkuus on vähintään SIMSCRIPT:in luokkaa ja sen algoritmiset ominaisuudet ovat hyvät.

Kaikkia näitä simulointikieliä on käytetty laajalti sekä teollisten että kaupallisten ongelmien simuloinnissa, mutta niitä on käytetty melko vähän liikennetutkimuksissa. GPSS:n käyttöä risteysalueiden simulointeihin on kirjoitettu artikkelissa /6/.

Algoritmisilla kielillä - FORTRAN, ALGOL, PL/1 - on ratkaisevia etuja erikoissimulointikieliin verrattuna: konetehokkuus, yhteensopivuus eri koneiden välillä ja ennen muuta niiden muunneltavuus probleeman mukaan /40/. Liikennesimulointeja tarkasteltaessa ehkä vakavin erikoissimulointikielten puute on se, että niitä käyttämällä on vaikea saada yksilöllistä tietoa ajoneuvosta tai ajajasta.

Se, että tämä simulointimalli päätettiin ohjelmoida FORTRAN-kielellä, johtui edellä kuvattujen algoritmisten kielten etujen lisäksi siitä, että FORTRAN on laajalti tunnettu ja osattu kieli, jonka tämän simuloinnin toimeksiantajan käyttäjien oletetaan ainakin hallitsevan. Lisäksi FORTRAN:in yksi etu, joka saat-
taa olla ohjelman kohdalla joskus ratkaisevakin. Sitä voi-
daan käyttää muistin koon suhteen pienemmissä koneissa kuin erikoissimulointikieliä. Ohjelmointikielen valinnalla ei haluttu tehdä mahdottomaksi sitä vaihtoehtoa, että varsinaiset simulointiajot suoritettaisiin pientietokoneella.

4.2

Simulointilogiikan pääpiirteet

Konstruoidun simulaattorin kokonaislogiikka on esitetty kaava-
maisesti kuvassa 4-3*, josta näkyy pää- ja aliohjelmien sekä
tiedostojen keskinäinen riippuvuus. Modulaarinen kokonaisoh-
jelma sisältää sekä sisäisiä että ulkoisia aliohjelmia, jotka
puolestaan voidaan vielä jakaa pienemmiksi ohjelmaosiksi.
Niitä ei kuitenkaan enää ole muodollisesti järjestetty alioh-
jelmiksi. Tällä tavoin mallin ohjelmointi on mahdollista toteut-
taa siten, että eri osamallit voidaan ohjelmoida toisistaan
riippumattomina, testata ja korjata liikuttamatta samalla koko
simulaattoria.

Simulointiohjelma jakautuu pääohjelmaan ja 12 FORTRAN V:n itsenäi-
seen aliohjelmaan. Jokainen pääohjelman kutsuma ohjelmayksikkö
(paitsi LUKU ja INKONF) käsittelee joka aika-askel risteyksessä
olevat ajoneuvot ja suorittaa niille "alaansa" kuuluvat operaai-
ot. Tehtävät on jaettu aliohjelmien kesken siten, että kunkin
aliohjelman suorittamat tehtävät muodostavat loogisen kokonaisu-
den.

Yleensä ns. teknis-matemaattiset ohjelmat ja erikoisesti simuloin-
tiohjelmat muodostuvat toimiviksi systeemeiksi vasta, kun niiden
kaikki komponentit (aliohjelmat) on koossa. Aivan samoin tämäkin
simulaattori vaatii toimiakseen kaikki aliohjelmat (tulostusohjel-
ma poisluettuna), sillä useimpien aliohjelmien input-tieto on

*Kuvat 4-3...4-22 on sijoitettu tutkimuksen liiteosaan (kohta 7.2)

edellisten aliohjelmien output-tietoa. Simulaattoria voisi vaikka verrata prosessiteollisuuden tuotantolinjaan, jossa yhdenkin mutterin puuttuminen aiheuttaisi koko prosessin katkeamisen.

4.3

Ohjelman osien tehtävät ja itsenäisyys

Seuraavassa on kustakin ohjelmayksiköstä erikseen kuvattu niiden tehtävät ja ohjelmointimenetelmä, mikäli se on jotenkin poikkeuksellinen tai muuten mainitsemisen arvoinen. Kustakin ohjelmayksiköstä on taulukon muodossa kuvattu se, miten tiedonsiirto tapahtuu sen ja muun ohjelmiston välillä argumenttiluettelon tai COMMON-alueiden välityksellä ja sen riippuvuus toisista ohjelmayksiköistä. Lisäksi näissä taulukoissa on luetteloitu ohjelmayksikön sisäisten aliohjelmien viitteet ja DEFINE-proseduurit.

Kussakin ohjelmayksikössä käytetty simulointilogiikka on kuvattu kulkukaavioina, joihin seuraavassa selostuksessa esiintyvät kuvaviittaukset kohdistuvat ja jotka on sijoitettu tekstiosan jälkeen liitteiden joukkoon. Kulkukaavioiden piirtäminen on suoritettu VTT:n CALCOMP-rumpupiirturilla käyttäen hyväksi kulkukaavion piirrosrutiinia FLOCT. Kulkukaavioissa olevat numeroviitteet tarkoittavat ohjelmakoodauksessa käytettyjä osoitteita. Kulkukaaviossa on käsittelylogiikka esitetty verrattain yksityiskohtaisesti, jolloin ohjelmistoon myöhemmin perehdyttäessä siitä saadaan tarkka kuva menemättä koodin tasolle ja mahdollisesti tehtävät muutokset on helposti paikallistettavissa.

4.3.1

Pääohjelma

Pääohjelma (kuva 4-4) toimii koko simulointisysteemin "isäntä"-ohjelmana ohjaten simuloinnin kulkua aliohjelmasta toiseen. Lisäksi siinä on logiikka, jolla lämmittelyvaiheen jälkeen suoritetaan yhteenvetotilastojen initialisointi ja joka valvoo ajoittain suoritettavia tulostustehtäviä simulointikellon avulla.

Pääohjelman alussa määritellään PARAMETER-käskyllä ohjelman useimpien taulukkomuuttujien dimensiorajat. Aliohjelmissa taulukkomuuttujien dimensiorajat kuitenkin määritetään muunneltaviksi syöttötietoina luettavien datojen avulla. Tällä "dynaamisella" järjestelyllä muistitila on tehokkaasti käytettävissä kulloinkin simuloitavana olevan risteyksen koon mukaan. Pääohjelma lukee tulostus- ja simulointiparametrit ja risteyksen dimensioidot sekä joitakin risteyksen mittoja (esim. ajokaistaleveyden). Ohjelma lisäksi vertaa luettuja dimensioidatoja PARAMETER-käskyllä määriteltäviin dimensiorajoihin. Jos jokin dimensioraja osoittautuu liian pieneksi, pääohjelman sisäinen aliohjelma YLITYS antaa virheilmoituksen ja ajo päättyy.

Risteyksestä on saatavissa hetkittäinen "koodikuva", josta ilmenee ajoneuvojen dynaamiset ominaisuudet ja liikennevalojen tila eri suunnissa. Johtamalla tämä "koodikuva" näyttöpäätteelle on mahdollista saada koko simuloidun risteyksen liikenne jatkuvana kuvaesityksenä näkyville. Tämä osa ohjelmasta on DELETE-käskyn avulla jätetty kääntämättä, mutta se saadaan haluttaessa mukaan muuttamalla DELETE-käskyn parametri.

Taulukossa 4-1 on kuvattu, miten pääohjelman muun ohjelmiston välinen tiedonsiirto tapahtuu.

4.3.2

Aliohjelma LUKU

Tämä aliohjelma (kuva 4-5) hoitaa risteyksen geometrinen rakennetta koskevien syöttötietojen ja valinnaisesti liikennevalo-ohjauksen syöttötietojen sisäänluvun sekä käyttäessä syöttötietojen tulostusoptiota formatoi ja kirjoittaa ne raporttiin. Ohjelma myös tarkistaa luettavien datakorttien lukumäärän ja niiden syöttötietojen oikeellisuuden, joiden kohdalla tarkistus oli helposti toteutettavissa.

Ajoreitti muodostuu ajouraluettelosta (koodina URAT), jossa ajourat ovat ajojärjestyksessä kuuden bitin pituisina kent-

tinä. Taulukkoon URAT ajouraluettelo pakataan FLD-funktion avulla tiivistetyksi tiedoksi. Kutakin reittiä kohti on varattu vain yksi tietokonesana ajouraluettelo varten taulukossa URAT, jolloin ajoreitti voi olla korkeintaan kuuden ajouran pituinen ilman ohjelman modifiointia (vrt. taulukon KONURA rakenteeseen).

Taulukoihin EURA ja SURA, joiden käyttöä on kuvattu kohdassa 3.3.3, luetaan alkioittain suoraan jatkuvan ajouran ja kääntyvän ajouran indeksit käyttäen apuna FLD-funktioita. Taulukon EURA (edeltäjäajoura) alkioista on vasemmalta lukien kuusi ensimmäistä bittiä varattu suoraan jatkuvan ajouran indeksille ja kuusi seuraavaa bittiä kääntyvän ajouran indeksille. Jos jotakin ajouraa ei ole, on vastaava alkion kentän arvo nolla. Samoin taulukon SURA (seuraaja-ajoura) alkioissa on kaksi kuuden bitin pituista kenttää, joista ensimmäisessä on suoraan seuraavan ajouran indeksi ja toisessa kääntyvän ajouran indeksi.

Ajourien lukumäärä voi korkeintaan olla 63 eli kuusinumeroisella 2-järjestelmän luvulla suurin ilmaistavissa oleva luku. Syöttötiedot on kuvattu tarkemmin kohdassa 4.7.1 ja niistä muodostetut tietueet kohdassa 4.6.

Taulukossa 4-2 on kuvattu, miten aliohjelman LUKU ja muun ohjelmiston välinen toedonsiirto tapahtuu.

4.3.3 Aliohjelman INKONF

Tällä aliohjelmalla (kuva 4-6) luetaan konfliktipistedatat ja muodostetaan niistä konfliktipistekohtaiset tiedot taulukkomuuttujaan KONFLP ja urakohtaiset konfliktipisteluettelot taulukkomuuttujaksi KONURA.

Kukin konfliktipistetietue sijoitetaan yhteen 36 bitin tietokonesanaan käyttämällä hyväksi FORTRAN V:n sisäistä funktiota FLD, jonka avulla tietokonesana voidaan jakaa kenttiin. Konfliktipistekohtaiset tiedot ovat taulukossa KONFLP, jonka

vasemmalta lukien kahdessa ensimmäisessä kuudesosasanassa on leikkaavien ajourien identifiointikoodi ja kaksi seuraavaa 7 bitin pituista kenttää on varattu konfliktipisteen etäisyyksien ilmaisemiseen vastaavien ajourien alusta metreinä. Seuraava kolmen bitin kenttä ilmaisee sektorin, jossa leikkaavat ajourat kohtaavat toisensa konfliktipisteessä ensimmäisen ajouran suhteen ja kolmen seuraavan bitin kentässä on vastaava tieto jälkimmäisen ajouran suhteen. Neljän seuraavan bitin kentän tieto on "pimeän kulman", näköesteen etäisyys ajouraa pitkin konfliktipisteestä lukien. Kenttiin jaosta on tarkempi kuvaus kohdassa 4.6.

Kussakin taulukon KONURA alkiossa on luetteloitu taulukon indeksiä vastaavalla ajouralla olevien konfliktipisteiden identifiointikoodit ajouran alusta lukien 7 bitin pituisina kenttinä. Jos joillakin ajourilla on useampia kuin 5 konfliktipistettä, käytetään jatkoluetteloita, jotka on indeksoitu ajourien lukumäärää suuremmilla indekseillä ja joita vastaavat indeksit on talletettu perusluettelon 5. kenttään. Jos siis perusluettelon 5. kentän tieto on suurempi kuin konfliktipisteiden lukumäärä, on jatkoluettelon indeksin arvo laskettavissa ko. 5. kentän tiedosta. Konfliktipisteiden lukumäärä voi olla korkeintaan 126 eli suurin 7 bitillä ilmaistavissa oleva 10-järjestelmän luku.

Lukurutiini tarkistaa konfliktipistedatakorttien lukumäärän. Käytettäessä syöttötietojen tulostusoptiota saadaan taulukkomuuttujat KONFLP ja KONURA tulostusraporttiin mukaan.

Taulukossa 4-3 on kuvattu, miten aliohjelman INKONF ja muun ohjelmiston välinen tiedonsiirto tapahtuu.

4.3.4 Aliohjelma PAIVIT

Aliohjelma (kuva 4-7) suorittaa systeemin päivityksen tasaaika-askeleittain. Se huolehtii ajansiirrosta ja laskee

ajoneuvoille uudet nopeuksien ja paikkakoordinaattien arvot aliohjelmassa KIIHTY tai KONFLI yksi reaktioaikayksikkö sitten laskettujen kiihtyvyysarvojen perusteella. Ohjelma lisäksi tarkistaa, ettei ajoneuville laskettu uusi paikka ole minimivälimatkaa lähempänä edeltäjää, mutta jos näin on, korjataan muualla laskettua kiihtyvyysarvoa siten, ettei minimivälimatkaa alitu. Minimivälimatkaksi lasketaan 80 % edeltäjäajoneuvon ns. efektiivisestä pituudesta ja ajoneuvon paikkakoordinaattihan ilmaisee ajoneuvon etuosan sijainnin.

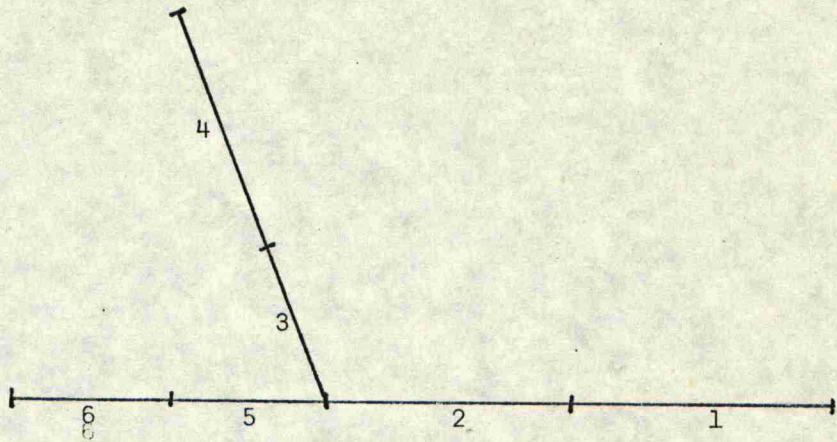
Aliohjelma päivittää myös taulukkomuuttujan XVALI, jossa on kunkin ajoneuvon kohdalla etäisyys edeltäjäajoneuvoon.

Aliohjelma käsittelee risteyksen ajouraketjuittain, jolloin ketjun ensimmäisenä ajourana on ulosmenokaistan identifiointikoodin mukaan nousevassa järjestyksessä seuraavana oleva ajoura.

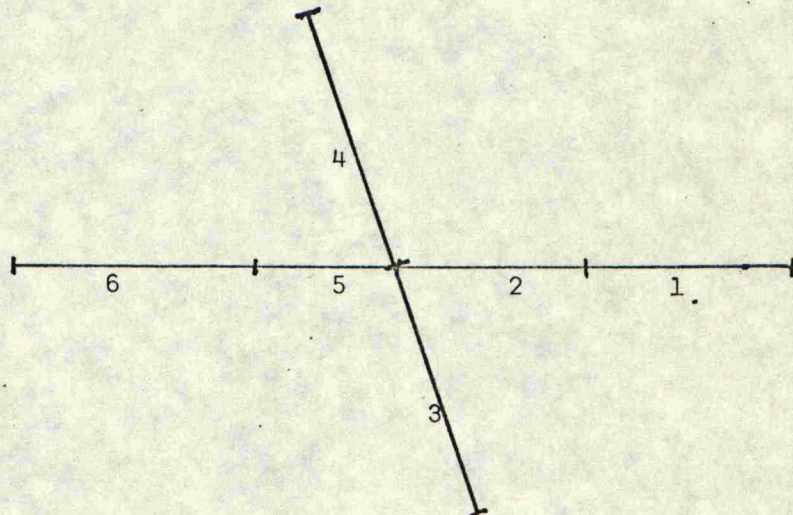
Ajourien liittymispisteeseen mahdollisesti tuleva kääntyvä ajoura ja sen suora seuraajaura (jos sellainen on) käsitellään rinnan suoran pääajouraketjun kanssa, jollei ko. kääntyvä ajoura ole jonkin toisen suoran pääajouraketjun osa.

Kuva 4-1 havainnollistaa ajourien käsittelyjärjestystä ja käsittelyn rajoitusta. Kuvan ajourat käsitellään identifiointikoodien mukaan nousevassa järjestyksessä. Ajourille annetut identifiointikoodit eivät esimerkissä ole kohdassa 3.3.1 esitettyjen standardien mukaiset. Ajouran 2 käsittelyn jälkeen käsitellään ajourat 3 ja 4. Ajouran 4 käsittely kuitenkin edellyttää, että se on ajouran 3 suora seuraaja-ajoura, eikä sillä saa olla minkäänlaista seuraaja-ajouraa. Jos ajoura 3 olisi ollut jonkin muun ajouraketjun osa, esimerkiksi kuvassa 4-2, saa ajouralla 4 olla minkälaisia seuraaja-ajouria tahansa. Kuvassa 4-2 olevien ajourien käsittelyjärjestys olisi nyt 1, 2, 5, 6, 3, 4.

Ajoneuvot puolestaan käsitellään alkaen



Kuva 4 - 1. Kääntyvän ajouran käsittelyjärjestys



Kuva 4 - 2. Ajourien käsittelyjärjestys

lähinnä käsiteltävänä olevan ajouran loppupäätä olevasta ajoneuvosta (koodina EAURA) kohti ajouran alkupäässä olevaa ajoneuvoa (VAURA), sillä edeltäjäajoneuvon informaatio vaikuttaa seuraaja-ajoneuvon päätöksentekoon ja käyttäytymiseen.

Taulukossa 4-4 on kuvattu, miten aliohjelman PAIVIT ja muun ohjelmiston välinen tiedonsiirto tapahtuu.

4.3.5 Aliohjelma POISTU

Aliohjelma (kuva 4-8) tutkii kullakin ulosmenokaistalla ensimmäisenä olevan ajoneuvon sijainnin poistumispisteen suhteen. Jos ajoneuvo on ohittanut poistumispisteen, poistetaan se ajouralta ja suoritetaan tarvittavat muutokset taulukkomuuttujiin SAUTO, EAURA, VAURA, NRPOIS.

Jos poistetun ajoneuvon seuraaja-ajoneuvo on noudattanut edeltäjän seuraamismallia, oletetaan sen kulkevan loppumatkansa tasaisella nopeudella epärealististen loppukiihdytysten välttämiseksi ajoneuvon nyt alkaessa pyrkiä tavoitenopeuteensa. Todellisuudessaan edeltäjäajoneuvo ei noin vain häviä edestä.

Jälkikäsitteilyä varten talletetaan tiedostoon auton identifiointikoodi, ajoreitti, matkan varrella sattuneiden konfliktien sijainnit, keskimääräinen nopeus ja pysähdysten kesto aika.

Taulukossa 4-5 on kuvattu, miten aliohjelman POISTU ja muun ohjelmiston välinen tiedonsiirto tapahtuu.

4.3.6 Aliohjelma URVAIH

Aliohjelma (kuva 4-9) käsittelee risteyksen ajourittain ajourien identifiointikoodin nousevan järjestyksen mukaan. Kultakin ajouralta haetaan ensimmäisenä oleva ajoneuvo ja jos sen ko. ajouralla kulkema matka on pidempi kuin ko. ajouran pituus, haetaan ajoneuville seuraavaksi tuleva ajoura reittiluettelosta.

Käsiteltävä ajoneuvo saa uuden ajouran mukaisen tavoite-nopeuden. Jos sen ajoreitin on uuden ajouran jälkeen määrä kääntyä, merkitään ajoneuvo kääntyväksi. Jos ajoneuvon uusi ajoura jatkuu edellisestä suoraan, tehdään muutokset vain taulukkomuuttujiin EAURA ja VAURA. Jos taas ajoneuvo on joutunut kääntymään uudelle ajouralle, suoritetaan vanhan suuntaisen liikennevirran uudelleenketjutus tekemällä muutokset taulukkomuuttujiin EAUTO, SAUTO, EAURA ja VAURA; lisäksi on uudesta suunnasta etsittävä edeltäjä- ja seuraaja-ajoneuvot ja ajoneuvo ketjutetaan uuteen liikennevirtaan muuttamalla taulukkomuuttujien EAURA, VAURA, EAUTO, SAUTO sisältöä taulukon 3-2 mukaisesti. Ajoneuvon kääntyessä ja liittyessä toiseen liikennevirtaan on laskettava myös etäisyys edeltäjäajoneuvoon.

Taulukossa 4-6 on kuvattu, miten aliohjelman URVAIH ja muun ohjelmiston välinen tiedonsiirto tapahtuu.

4.3.7

Aliohjelma GENERO

Aliohjelma (kuva 4-10) hakee ensin ne sisääntulokaistat, joiden generointipisteeseen on generoitu ajoneuvo. Generointipisteet käsitellään sisääntulokaistojen identifiointikoodin mukaisessa nousevassa järjestyksessä.

Jos simulointikelloaika on kulunut generoidun ajoneuvon sisääntulohetken yli, sovitetaan ajoneuvo ajouralleen ja tehdään merkinnät taulukkoihin EAUTO, SAUTO, VAURA. Jos se ei sovi ajouralle, lasketaan seuraava sisääntulohetki, joka on aika-askeleen kuluttua. Ajouralleen sovitetun ajoneuvon ja sen seuraaja-ajoneuvon välinen aikaväli generoidaan kaksihuippuisesta eksponenttijakautumasta simulointimallin kohdan 3.8 mukaisesti. Tälle uudelle ajoneuville generoidaan myös sen pysyvät ominaisuudet: identifiointikoodi, (joka on NRPOIS-tilukon ensimmäinen vapaana oleva koodi), reitti (,joka määräytyy tunnetusta generointipisteestä ja jos on

useita vaihtoehtoisia reittejä, arpomisen tuloksesta), ajoneuvon tyyppi, tavoitenopeus, haluttu jarrutusnopeus, maksimikiikkyvyys ja ajoneuvon pituus. Ohjelma laskee vielä äsken generoidulle ajoneuvolle liiketilan aika-askelen päättyessä.

Taulukossa 4-7 on kuvattu, miten aliohjelman GENERO ja muun ohjelmiston välinen tiedonsiirto tapahtuu.

4.3.8 Aliohjelma LVALOT

Aliohjelma LVALOT (kuva 4-11) hoitaa kiinteäaikaisen valohjauksen päivityksen ja tutkii, onko aika vaihtaa valoja. Valotilataulukon päivitys tapahtuu lisäämällä kunkin suunnan kestoajaa yhdellä aika-askelen suuruudella yksiköllä. Jos jonkin suunnan kestoaja ylittää taulukkomuuttujassa VAIHE ko. suuntaan ilmoitetun kestoajan, suoritetaan liikennevalojen vaihtaminen ko. suuntaan. Jokaisella sisään-tulokaistalla on taulukossa LVALO alkionsa, joka ilmaisee ko. suuntaan osoittavan liikennevalokoodin. Lopuksi ohjelma etsii seuraavan uudelleenvaihtojankohdan ja testaa, onko se jo nyt käsillä.

Taulukossa 4-8 on kuvattu, miten aliohjelma LVALOT ja muun ohjelmiston välinen tiedonsiirto tapahtuu.

4.3.9 Aliohjelma JONOT

Aliohjelmalla (kuva 4-12) kerätään tietoa sisään-tulokaistoilla esiintyvien liikkumattomien jonojen pituuksista. Tällä tavalla saadaan helposti käsitys risteyksen välityskyvystä. Aliohjelma etsii kultakin sisään-tulokaistalta pysähtyneet ajoneuvot. Jos niiden lukumäärä on suurempi kuin vastaavan jonon talletettu pituus, kasvatetaan jonoa yhdellä. Heti kun jonon ensimmäinen ajoneuvo lähtee liikkeelle, katsotaan vastaavan jonon häviävän. Kunkin jonon pituus on tiedossa joka aika-askel. Aina, kun jono häviää,

viedään maksimijonon pituus kirjanpito-ohjelman ENTRY kohtaan KIRJA3.

Taulukossa 4-9 on kuvattu, miten aliohjelman JONOT ja muu ohjelmiston välinen tiedonsiirto tapahtuu.

4.3.10

Aliohjelma KIIHTY

Aliohjelmalla KIIHTY (kuva 4-13) yhdessä sen sisäisissä ohjelmissa ja muissa aliohjelmassa olevien kiihtyvyyden laskemisrutiinien kanssa lasketaan ajoneuvoille kiihtyvyydsarvot. Laskemisen perusteena on ajoneuvojen nopeuden ja aseman sen hetkinen tila risteyksessä, jolloin rutineissa tehtävät päätökset alkavat vaikuttaa ajoneuvojen liiketilaan vasta yhden viiveaikayksikön kuluttua.

Aliohjelmalla hoidetaan yleinen kiihtyvyyden laskeminen normaalisti eteneville ajoneuvoille ja määrätään ja ohjataan siirtyminen ajajan käyttäytymistä simuloiviin alirutiineihin, joita käytetään erikoistilanteissa kuten keltaista valoa lähestyttäessä, punaista valoa lähestyttäessä, käännyttyäessä toiselle ajouralle ja risteävän ajoneuvon tai jalankulkijan aiheuttaessa konfliktin. Erikoistilanteiden käsittely tapahtuu aliohjelman KIIHTY alirutiineilla, VIHKEKEL, PUNKEL, PUNLAH, KAANTY, jotka ovat muodollisesti FORTRAN V:ssä määriteltäviä sisäisiä aliohjelmiä sekä aliohjelmalla KONFLI, joka on muodollisesti aliohjelman KIIHTY ulkoinen aliohjelma (itse asiassa looginen funktio).

Kiihtyvyyden laskemismallien antamista eri tuloksista ajoneuvon käytettäväksi valitaan arvoltaan pienin eli ajoneuvot etenevät rajoittavimman mallin mukaan.

Aliohjelmalla KIIHTY lasketaan kiihtyvyydsarvot niille ajoneuvoille, joiden liikettä ei rajoita muu kuin edessä oleva ajoneuvo. Tällöin ajoneuvon kiihtyvyys lasketaan jonkin seuraavana olevan etenemismallin mukaan: vapaa eteneminen, edeltä-

jän seuraaminen, pysähtyvä eteneminen tai pysähtynyt tila. Aliohjelma KIIHTY käsittelee risteyksen ajourat ja ajoneuvot täsmälleen samassa järjestyksessä kuin kohdassa 4.3.4 selostettu aliohjelma PAIVIT.

Jos risteys on valo-ohjattu, aliohjelma KIIHTY tarkistaa ajoneuvon sijainnin seis-viivaan nähden, joka on jonkin verran liikennevalo-opastimia lähempänä risteystä (luettu syöttötietona). Jos ajoneuvo ei ole ylittänyt seis-viivaa, siirretään käsittely sopivaan valo-ohjausaliohjelmaan. Jos taas ajoneuvo on ylittänyt seis-viivan, tutkitaan KONFLI-aliohjelmalla, onko ajoreitti esteetön. Jos ajoreitti on esteetön, ajoneuvolle ruvetaan laskemaan kiihtyvyyssarvoa. Jos se etenee kääntyen, kiihtyvyyssarvo lasketaan alirutiinilla KAANTY, muutoin laskennan suorittaa KIIHTY.

Ensiksi KIIHTY tutkii, onko käsiteltävänä oleva ajoneuvo riippumaton edeltäjäajoneuvostaan. Riippumattomuus katsotaan saavutetuksi, jos ajoneuvon välimatka edeltäjään on syöttötietona luettua parametriä suurempi tai jos suhteellisen nopeuden muutoksen havaitsemisen kynnys (kaava 3-11) ei ylity. Riippumatolle ajoneuvolle lasketaan kiihtyvyyssarvo vapaan etenemismallin mukaan. Edeltäjäajoneuvosta riippuvalle ajoneuvolle kiihtyvyyssarvot lasketaan edeltäjän seuraamismallin mukaan, jollei näin vapaan etenemismallin antamaa kiihtyvyyssarvoa ylitetä.

Jos edeltäjäajoneuvo on päättänyt alkaa pysähtyä tai on jo pysähtynyt, käsiteltävänä oleva ajoneuvo noudattaa pysähtymismallia tai sekin on jo pysähtynyt. Aliohjelman ORJAK antaa normaalijakautuneita satunnaislukuja käytetään pysähtymiskiihtyvyyden lausekkeessa (kaava 3-12) pysähtymispaikkavirheenä.

Taulukossa 4-10 on kuvattu, miten aliohjelman KIIHTY ja muun ohjelmiston välinen tiedonsiirto tapahtuu.

Alirutiini VIHKE

Alirutiinilla (kuva 4-14) määrätään ajoneuvon käyttäytyminen, kun sen edessä oleva liikennevalo on vaihtunut vihreästä keltaiseksi. Jos ajoneuvo päättää pysähtyä, kontrolli siirtyy punaista lähestyminen-rutiinille. Jos ajoneuvo päättää jatkaa ajoa, kontrolli siirtyy yleisen kiihtyvyyden laskemismallille. Päätös tehdään satunnaisprosessilla hidastuvuuden funktiona olevaan pysähtymistodennäköisyystaulukkoon nojautuen sen jälkeen, kun on generoitu tavoitepysähtymispaikka ja laskettu pysähtymishidastuvuus. Satunnaisprosessissa käytetään välille (0,1) tasan jakautuneita satunnaislukuja.

Alirutiini PUNKEL

PUNKEL-rutiinia (kuva 4-15) sovelletaan, kun ajoneuvon edessä oleva liikennevalo on muuttunut punaisesta keltaiseksi. Jollei ajoneuvoa uhkaa konfliktitilanteeseen joutuminen sen lähtiessä liikkeelle jo keltaisen valon palaessa, ajoneuvo lähtee liikkeelle todennäköisyydellä 0,5, muuten siirrytään PUNLAH-alirutiiniin. Arpomisessa käytetään tasanjakautuneita satunnaislukuja.

Alirutiini PUNLAH

Alirutiinilla (kuva 4-16) lasketaan kiihtyvyyssarvot kaikille niille ajoneuvoille, jotka ovat niillä sisääntulo-kaistoilla, joihin suuntiin palaa punainen liikennevalo.

Rutiini käsittelee ensimmäisen ajoneuvon edellisestä riippumattomana ja sille määrätään jonkin seuraavista käyttäytymismalleista. Jos se on pysähtynyt kauaksi seis-viivasta (ja edeltäjää ei ole) kuljetetaan se lähemmäksi seis-viivaa vapaan etenemismallin mukaan, seis-viivalle pysähtynyt ajoneuvo jää pysähtyneeksi. Jos ajoneuvolle on jo generoitu tavoitepysähtymispaikka ja laskettu pysähtymishidastuvuus,

noudattaa se pysähtyvän etenemisen mallia. Jos ajoneuvo ei aikaisemmin ole päättänyt ruveta pysähtymään, rutiini tutkii, alkaako se nyt pysähtyä generoimalla tavoitepysähtymispaikan ja sen mukaisen pysähtymishidastuvuuden. Mikäli pysähtymishidastuvuus on haluttua jarrutushidastuvuutta pienempi, ajoneuvo alkaa noudattaa pysähtymismallia.

Samalla sisääntulokaistalla olevat seuraaja-ajoneuvot noudattavat vapaan etenemisen mallia, edeltäjän seuraamismallia tai pysähtymismallia riippuen edeltäjäajoneuvon etäisyydestä ja sen noudattamasta etenemismallista. Päätökset tehdään samanlaisen algoritmin avulla kuin aliohjelmassa KIIHTY yleisen kiihtyvyyden laskemiserutiinia käytettäessä.

Alirutiini KAANTY

Alirutiini (kuva 4-17) määrää kääntyville ajoneuvoille kiihtyvyyssarvot kaavan 3-12 mukaan, kun ne ovat tietyn matkan päässä kääntymiskohdasta. Tämä matka on luettu parametriksi ja se on yhteinen kaikille kääntyville ajoneuvoille koko risteyksen alueella. Kääntymiskiihtyvyyden laskentarutiinia aletaan soveltaa, kun ajoneuvo seuraavalle ajouralle siirtyessään kääntyy. Heti, kun kääntymisjarrutushidastuvuus on pienempi kuin ajoneuvolle ominainen haluttu jarrutushidastuvuus, ajoneuvo saa kiihtyvyyssarvokseen kääntymisjarrutushidastuvuusarvon. Tätä hidastuvuusarvoa ajoneuvo noudattaa kääntymiskohtaan asti, jossa sen nopeus on lähellä parametriksi luettua kääntymisnopeuden arvoa. Kun ajoneuvo on juuri kääntymässä, lasketaan sille kääntymiskiihtyvyys kaavasta 3-15.

Kun rutiinilla on laskettu kääntymiskiihtyvyyssarvo, tarkistetaan aiheuttaako edessä oleva ajoneuvo edeltäjän seuraamismallia noudatettaessa pienemmän kiihtyvyyssarvon kuin edellä laskettiin. Jos näin on, ajoneuvolle annetaankin tämä kiihtyvyyssarvo.

4.3.11

Aliohjelma KONFLI

Aliohjelma KONFLI (kuva 4-18) on muodoltaan FORTRAN V:n looginen funktio, joka saa arvot TRUE ja FALSE. Aliohjelma KIIHTY ja alirutiini PUNKEL kutsuvat funktiota KONFLI. Jos KONFLI:n arvo on TRUE, on käsiteltävänä olevalle ajoneuvolle annettu kiihtyvyyssarvo laskettu funktiossa KONFLI. Jos KONFLI:n arvo on FALSE, määrätään ajoneuvon kiihtyvyyssarvo aliohjelmassa KIIHTY tai sen alirutiinissa.

Funktiota KONFLI sovelletaan kerrallaan vain funktion kutsussa argumenttina olevan identifiointikoodin ilmaisemaan ajoneuvoon. Konfliktin mahdollisuutta tutkitaan ajoneuvon reitillä olevissa konfliktipisteissä. Kerrallaan ajoneuvon edestä tutkittavien konfliktipisteiden lukumäärä ilmoitetaan PARAMETER-käskyllä funktion alussa. Ohjelma etsii ensin käsiteltävänä olevan ajoneuvon edestä lähinnä olevan konfliktipisteen. Jos tätä konfliktipistettä lähestyy samasta suunnasta myös jokin toinen konfliktissa oleva edeltäjäajoneuvo siirretään tämän ajoneuvon käsittely yleisen kiihtyvyyden laskemiseksi rutiinille. Kun on löydetty konfliktipiste, seuraavaksi etsitään risteävältä ajouralta ajoneuvo, joka on juuri konfliktipisteessä tai sitä lähestymässä. Jos risteävä ajoura on jalankulkijain suojatie, siirrytään alirutiiniin JALKNF. Jollei risteävältä ajouralta löydy ajoneuvoa tutkitaan vielä seuraaja-ajoura, jos sieltä löytyisi ajoneuvo.

Kun on löytynyt kaksi samaa konfliktipistettä eri suunnista lähestyvää ajoneuvoa, lasketaan molempien etäisyys konfliktipisteeseen alifunktiolla XTOD ja kohdassa 3.6.3 esitetyn nopeuden ja matkan arvioimismallin avulla määritetään käsiteltävänä olevan ajoneuvon molemmille ajoneuvoille arvioimat ajoajat ja etäisyydet konfliktipisteeseen. Jos risteävälle ajoneuvolle arvioitu ajoaika on paljon (ohjelmassa viisi kertaa) suurempi kuin käsiteltävänä olevan ajoneuvon ajoaika, ei ole konfliktin uhkaa ainakaan ko. konfliktipisteessä. Muussa tapauksessa tutkitaan havaitsemista satunnaisprosessilla.

Havaitsemistodennäköisyysjakatuma käytetään jatkuvaa funktiota, joka riippuu käsiteltävänä olevan ajoneuvon kulkusuunnan suhteen mitatun havaitsemiskulman puolikkaasta \cos -funktion mukaan ja molempien ajoneuvojen etäisyydestä konfliktipisteeseen negatiivisen eksponenttifunktion mukaan (kaava 3-14). Havaitsemiskulma on helposti laskettavissa \cos -väittämän mukaan, kun tiedetään ajoneuvojen etäisyydet konfliktipisteeseen ja ajourien välinen kohtauskulma. Ajourat on aina approksimoitavissa riittävällä tarkkuudella suoriksi. Mikäli jakautumasta TASAN generoidulla satunnaisluvulla arpomalla päädytään siihen, että havaitseminen on tapahtunut, ajoneuvo reagoi toiseen riippuen niiden välisestä etuajo-oikeudesta.

Ajoneuvojen välinen etuajo-oikeus määräytyy niiden kulkevien ajourien kohtaus suunnasta toisiinsa nähden. Kohtaus suunta on luettu syöttötietona taulukkoon KONFLP.

Loogisella alifunktiolla HYVVAL tutkitaan, päättääkö ajoneuvo lähestyä edelleen konfliktipistettä vai alkaako se valmistautua pysähtymiseen. Jos välinhyväksymispäätös on kielteinen, generoidaan autolle tavoitepysähtymispaikka ja lasketaan sen mukainen pysähtymishidastuvuus. Etuajo-oikeutettu ajoneuvo alkaa noudattaa pysähtymishidastuvuutta vasta, kun pysähtymishidastuvuus on pienempi kuin kaksi kertaa haluttu jarrutushidastuvuus. Vastaavasti väistämisvelvollinen ajoneuvo alkaa noudattaa pysähtymishidastuvuutta heti, kun se on pienempi kuin haluttu jarrutushidastuvuus. Jos lasketut pysähtymishidastuvuudet eivät ole haluttua jarrutushidastuvuutta pienempiä, ajoneuvo siirtyy yleisen kiihtyvyyden laskemisruutiinien piiriin.

Alifunktion KONFLI tiedonsiirto kutsuvien aliohjelmien kanssa tapahtuu argumenttiluettelon välityksellä. COMMON-käskyillä hoidetaan taulukkomuuttujien dimensiomääritykset ja aliohjelmasta INKONF leikkaavien ajourien sektorijaotustiedot.

Taulukossa 4-11 on kuvattu, miten aliohjelman KONFLI ja muun ohjelmiston välinen tiedonsiirto tapahtuu.

Alirutiini HYVVAL

Tässä loogisessa alifunktiossa (kuva 4-19) määrätään satunnaisprosessilla eteneekö ajoneuvo konfliktipisteeseen, kun tiedetään risteävää ajouraa pitkin etenevän ajoneuvon arvioitu ajoaika ja käsiteltävänä olevan auton itsestään arvioima ajoaika. Jos käsiteltävänä oleva ajoneuvo ei ole aivan lähellä konfliktipistettä (ajoaika yli 3 sekuntia) hyväksyy se välin, jos risteävä ajoneuvon arvioitu ajoaika on vielä suurempi. Samoin se hyväksyy välin varmasti, jos risteävän ajoneuvon arvioitu ajoaika on suurempi kuin 10 sekuntia.

Alifunktio XTOD

Funktio (kuva 4-20) laskee argumentteina annettujen ajoneuvon sen hetkisen paikan ja konfliktipisteen välisen etäisyyden. Konfliktipisteen on oltava ajoneuvon reitillä. Jos ajoneuvo on jo ohittanut ko. konfliktipisteen, funktio XTOD saa negatiivisen etäisyysarvon. Muulloin sen saama arvo on positiivinen.

Alirutiini JALKNF

Tässä rutiinissa (kuva 4-21) käsitellään ajoneuvojen ja jalankulkijan väliset konfliktit. Jalankulkijat generoidaan konfliktipisteisiin käyttäen hyväksi todennäköisyysjakautumia, jotka riippuvat suojatiellä palavan liikennevalon laadusta, sen kestoajasta ja konfliktipisteen sijainnista kadun reunaan nähden. Jos liikennevalo-ohjaus ei ole päällä, jalankulkijan todennäköisyysjakautuman sijasta käytetään kullekin suojatielle omaa jalankulkijan todennäköisyyslukua. Ajoneuvo pysähtyy konfliktipisteessä jalankulkijan vuoksi, jos ajoneuvon pysähtymiseksi tarvittava jarrutushidastuvuus on haluttua jarrutus-
hidastuvuutta pienempi.

4.3.12

Aliohjelma KIRJA

Aliohjelmassa (kuva 4-22) hoidetaan keskitetysti risteys-systeemin simuloinnin aikaisten yhteenvetotilastojen laske-minen, formatointi ja kirjoittaminen. Ohjelma suorittaa tulostuksen simulointiajan lopussa ja korkeintaan 10 väli-tulostusta syöttötiedoissa määriteltynä ajanhetkinä. Tu-lostettavat tilastot ovat ajan suhteen kumulatiivisia.

Aliohjelma rakentuu useasta erillisestä apurutiinista, jois-ta osa on ENTRY-käskyn muodossa ja osa sisäisinä aliohjelmina.

1. ENTRY KIRJA2-rutiinin kutsut ovat aliohjelmassa KONFLI. Rutiini tutkii, onko tiettyjen ajoneuvojen välinen konf-liktitilanne tietyssä konfliktipisteessä esiintynyt jo edellisellä aika-askeleella ja jos on, mihin suuntaan sen vaarallisuusaste on muuttunut. Kukin konfliktitilanne lasketaan tilastoihin mukaan vain yhden kerran silloin, kun se on vaarallisimmillaan, vaikka aliohjelma KONFLI käsittelisikin samojen ajoneuvojen välistä konfliktitilan-netta useammankin peräkkäisen aika-askelen kuluessa.
2. Sisäinen aliohjelma TILAST tilastoi kustakin konfliktitilanteesta konfliktoivien ajoneuvojen identifiointikoo-dit ja niiden tyyppiluokat, konfliktipisteen identifi-oointikoodin, ajoneuvojen etäisyydet konfliktipisteeseen nähden, niiden nopeudet ja hidastuvuudet sekä konfliktien ajanhetken tiedostoon, jota voidaan simulointiajan jälkeen tutkia ja käsitellä esimerkiksi tilastollisin menetelmin. Lisäksi sisäinen aliohjelma TILAST laskee kunkin ajoneuvon kohdalla sattuneiden konfliktien lukumäärän ja tilastoi konfliktipisteet, laskee konfliktipistekohtaisesti konf-liktien lukumäärät, keskimääräiset tapahtumisaikavälit ja keskimääräiset ajoneuvojen nopeudet sekä jarrutushi-dastuvuusjakautumat. Ajoneuvoluokkien (myös jalankulki-jat mukana luokkana nro 6) väliset konfliktimääräjakau-tumat tilastoidaan koko risteysalueelta ja haluttaessa myös erikseen korkeintaan yhdeksästä syöttötietoina ilmoi-tetusta konfliktipisteestä.

3. ENTRY KIRJA3-rutiinin kutsu on aliohjelmassa JONOT. Rutiini laskee sisääntulokaistoittain keskimääräiset maksimijononpituudet.
4. ENTRY KIRJA4-rutiinin kutsu on pääohjelmassa. Rutiini tulostaa aliohjelman KIRJA muiden rutiinien laskemia kumulatiivisia tuloksia. Tulostustiedot on kuvattu tarkemmin kohdassa 4.7.3.
5. Sisäinen aliohjelma FREKVE etsii frekvenssitaulukon indeksin, jota vastaavaan alkioon argumenttina annettu luku sijoitetaan. Argumentteina annetaan myös lukusuoran osan päätepisteet, joiden välissä toimitaan, ja jaotus ko. välillä.

Taulukossa 4-12 on kuvattu aliohjelman KIRJA ja muun ohjelmistojen välinen tiedonsiirto.

4.3.13 Aliohjelma SATLUK

Aliohjelmalla SATLUK yhdessä varsinaisen satunnaislukugeneraattorin kanssa generoidaan tarvittavien todennäköisyysjakautumien mukaan jakautuneita lukuja.

Satunnaislukugeneraattorilla tuotetaan pseudo-satunnaislukuja kerto-kongruenssi-menetelmän mukaan, joka on Andersinin /2/ mukaan käytännössä useimmin esiintyvä. Menetelmän etuja ovat vielä hyvät statistiset ominaisuudet, pitkä jakso sekä nopeus ja helppo ohjelmoitavuus symbolisillakin kielillä. Menetelmä perustuu rekursioyhtälöön:

$$y_{i+1} = \alpha y_i + r \pmod{m} \quad (4-1)$$

Moduli m valitaan tavallisesti sopivaksi 2^k :n tekijäksi käyttettäessä binäärisiä tietokoneita.

Koska tietokone on äärellinen ei voida tuottaa ei-toistettavaa sarjaa lukuja ilman ulkoista "apua". Ahrensin /1/

mukaan kaikille kaavan 4-1 tuottamilla sarjoilla on äärellinen periodi. Jos r on pariton ja $\alpha = 1(\text{mod } 4)$, sarjan 4-1 periodiksi tulee $m = 2^k$. Jos $r = 0$, sarjan 4-1 periodi on ainakin $m/4 = 2^{k-2}$. Tämä arvo saavutetaan, jos alkupiste y_0 on pariton ja jos $\alpha = \pm 5(\text{mod } 8)$. Ahrens [1] olettaa, että $r = 0$, $m = 2$, $y_0 = 1(\text{mod } 4)$ ja $\alpha = 5(\text{mod } 8)$, jolloin kaikki muotoa $4n + 1$ olevat jäännökset esiintyvät sarjan 4-1 täydessä jaksossa täsmälleen kerran. Jos jakson pituudeksi otetaan vain $m/4$, ohjelman koodaus helpottuu. Se, että joukossa $\{y_i\}$ voi esiintyä nollakin, on hyödyksi eräissä satunnaislukutransformaatioissa.

Asettamalla $x_i = y_i/m$ saadaan rekursio 4-1 seuraavaan muotoon, kun $r = 0$:

$$x_{i+1} = \alpha x_i (\text{mod } 1); \alpha = 5(\text{mod } 8); x_0 = (4n+1)2^{-k} \quad (4-2)$$

Rekursio 4-2 käytölle on kaksi etua: ensiksi perusjakautumana pidetään useimmiten tasaista jakautumaa välillä $(0,1)$. Toiseksi kokonaislukuaritmetiikka rajoittuu tietokoneissa yhden muistisanan pituisiin sanoihin, jolloin muistin ylivuodon välttämiseksi olisi käytettävä pientä m :ää ja lyhyitä periodeja. Rekursio 4-2 käyttö lopullisessa algoritmossa mahdollistaa kaksoissanan käytön FORTRAN-ohjelmassa. Ahrens esittää artikkelissa [1] perinpohjaiset testitulokset kongruenssigeneraattorille, joten tässä työssä satunnaislukugeneraattoria ei ole erikseen testattu.

Satunnaislukugeneraattorin tuottamia tasan jakautuneita satunnaislukuja transformoidaan funktio-aliohjelmassa SATLUK. Eri jakautumatransformaatiot muodostavat ohjelmassa kukin oman ENTRY-kohtansa. Transformaatiot on toteutettu Mihranin [27] esittämien algoritmien mukaan. Ohjelmalla on mahdollista tuottaa

1. välille (A, B) tasan jakautuneita satunnaislukuja,
2. normaalijakautuneita satunnaislukuja, kun keskiarvo ja hajonta ovat parametreinä,

3. Poisson-jakautuneita satunnaislukuja,
4. eksponentti jakautuman mukaisia satunnaislukuja ja
5. hyper-eksponenttijakautuman mukaisia satunnaislukuja.

Taulukossa 4-13 on kuvattu aliohjelman SATLUK ja muun ohjelmavälinen tiedonsiirto.

4.4

Koneriippuvat osat

Ohjelmisto on koodattu UNIVAC 1100 sarjan FORTRAN V-kielillä, jossa on joitakin eroja verrattuna ANSI Standard FORTRANiin. Useimmat erot johtuvat FORTRAN V:n laajennuksista ANSI FORTRANiin verrattuna. Muut erot johtuvat joko siitä, että FORTRAN V:ssä on jokin rajoitus, jota ei ole ANSI FORTRAN:ssa tai siitä, että FORTRAN V ja ANSI FORTRAN tulkitsevat eri tavalla jonkin saman alkukielisen käskyn.

Käsiteltävänä olevassa ohjelmistossa ovat seuraavat ANSI Standard FORTRAN:sta poikkeavat ominaisuudet.

1. Ohjelmayksiköissä on käytetty sisäisiä aliohjelmiä ja alifunktioita.
2. Pääohjelmassa ja aliohjelmassa KONFLI on tietonimiä käytetty identifioimaan PARAMETER-muuttujia, joille on annettu kokonaislukuarvoja.
3. Hollerith-vakioiden esitysmuotona on käytetty muotoa 'hhh...h'.
4. Aliohjelmassa PAIVIT ja GENERO on Hollerith-vakioita käytetty aritmeettisessä sijoituskäskyssä.
5. Aliohjelmassa KIRJA on käytetty 4-dimensioista taulukkomuuttujaa KNFKES.
6. Aliohjelmissa LUKU (taulukkomuuttuja EPIT) ja POISTU (taulukkomuuttuja IAPU) on käytetty FORTRAN V:n sallimaa indeksilauseketta, jonka muotoa on $e = \alpha_1 M_1 \alpha_2 M_2 \dots \alpha_n M_n$, jossa α_i :t edustavat + tai - merkkiä ja M_i :t lauseketta k tai $k_1 \times k_2 \times \dots \times k_j$, jossa k_i :t ovat kokonaislukuvakioita tai kokonaislukumuuttujia.

7. On käytetty käskyjä ENTRY ja DEFINE, jotka ovat ANSI FORTRAN:ssa proseduurinimiksi kutsuttujen symbolisten nimien luokien laajennus.
8. Sisäisessä aliohjelmassa KAANTY ja KIRJA4 on käytetty RETURN-käskyä, jonka avulla paluu tapahtuu muuhunkin kutsuvan ohjelman käskyyn kuin vain kutsua seuraavaan käskyyn. Paluu tapahtuu argumenttiluettelon n:nnen argumentin määrittämään osoitteeseen.
9. FORTRAN V:n sallimaa vertailuoperaattorin käyttämistä kokonaislukulausekkeen ja reaalilukulausekkeen välillä on käytetty hyväksi.
10. FORTRAN V:n sisäisiä funktioita FLD on käytetty 36-bittisten pakattujen kenttien käsittelyssä.
11. Loogista IF käskyä on käytetty DO-luupin viimeisenä käskynä.
12. Joissakin DO-luupeissa indeksiparametri voi olla negatiivinen tai nollan arvoinen.
13. Lukukäskyissä on käytetty lisänä END = & lausetta, jossa & on siirtymisosoite, jos EOF havaitaan ennen lukemisen päättymistä.
14. COMMON-muuttujia on käytetty aliohjelmissa taulukkomuuttujien muunnettavina dimensiorajoina.
15. FORTRAN V:n trigonometriset funktiot vaativat radiaaneissa mitattuja kulmia.
16. Pääohjelmassa on käytetty DELETE-käskyä.

4.5

Ohjelman muuntelumahdollisuudet

Ohjelmalla simuloitavan risteyksen koon muuntelumahdollisuus on lähes täydellinen sekä ajoneuvojen lukumäärän että fyysisen risteyksen suhteen. Koska simuloitavan risteyksen geometrinen muoto luetaan koodeiksi syöttötietoina, voi ohjelmalla simuloida minkälaisia risteyksiä tahansa. Todellisina rajoituksina on nähtävissä vain tietokoneajoaika ja tietokoneen muistikapasiteetti, joista myöskään ei ole odotettavissa muodostuvan rajoituksia käytettäessä UNIVAC 1108:n kaltaista suurtietokonetta. Kohdassa 3.3 määritellyn katuristeyksen rakenteen vuoksi ajoneuvojen reittien ajourien maksimilukumäärärajoitus (6) on helposti muutettavissa ohjelman modifioinnilla, jossa ajoreittien sallitaan koostuvan perusajouraluettelon lisäksi jatkoajouraluettelosta, kuten taulukkomuuttujan KONURA kohdalla on jo toteutettu.

Koska simulointimalli rakentuu systeemin mikro-tasoisten komponenttien kuvaamisesta, simulaattori on modifioitavissa minkä tahansa risteysliikenneilmiön tutkimiseen sekä liikenteen suoritteiden ja risteyksen toimivuuden mittaamiseen (vrt. kohta 3.9).

Ohjelmassa käytetyt ajoneuvojen etenemismallit on esitetty DEFINE-proseduurin avulla yhtenä kokonaisuutena, jolloin malleissa esiintyviä parametrejä ja jopa koko malleja on helppo muunnella. Itse asiassa koko ohjelmoinnin ajan on pyritty pitämään mielessä ohjelman myöhemmät modifiointitarpeet, jonka vuoksi koko ohjelmiston määrittelyt ja dimensiorajat on koodattu siten, että niitten muuttaminen käy mahdollisimman joustavasti. Tämä on ollut toteutettavissa FORTRAN V:n PARAMETER-käsitteellä ja COMMON-käskyllä määriteltävissä olevilla dimensiorajoilla.

Simulaattorin tulostustoiminnot ovat vielä varsin karkeat, joka johtuu pääasiassa siitä, että koska mallia ei ole voitu kenttätutkimusten puuttumisen vuoksi kalibroida todelliseen

risteykseen, ei ole tiedostettu kaikkia mielenkiintoisia konfliktitilanteita kuvaavia karakteristisia suureita.

Risteyssimuloinnissa olisi varsin helposti toteutettavissa graafisen tietojenkäsittelyn hyväksikäyttö. Ohjelmassa lasketut ajoneuvojen paikkakoordinaatit jokaisella tarkasteltavalla hetkellä voitaisiin johtaa näyttölaitteelle, joka piirtäisi ao. paikkoihin identifioitavissa olevan ajoneuvon, jonka koko ja muutkin ominaisuudet voisivat olla näyttöpäätteen kuvassa mukana. Täten risteyksen dynaamista toimintaa voitaisiin seurata jatkuvana kuvana filmin tavoin.

4.6

Muuttujaluettelot

Käsitettä muuttuja käytetään tässä kappaleessa siinä merkityksessä kuin se on FORTRAN:in sääntöjen mukaan määritelty:

" FORTRAN muuttuja on sellaisen suureen symbolinen esitys, joka saa arvon. Arvo voi olla muuttumaton (ts. vakio) tai se voi muuttua ohjelman eri toteutuskertojen mukaan tai ohjelman sisäisen tilan mukaan". (IBM:system/360, FORTRAN IV Language s.13).

Muuttujat on jaettavissa neljään ryhmään sen mukaan, miten niitä käytetään:

- a) input-muuttujat (syöttötiedot)
- b) prosessimuuttujat (osa prosessimuuttujista voi olla syöttötietojakin)
- c) apumuuttujat (mm. DO-luoppien indeksit, taulukko-muuttujien indeksit)
- d) output-muuttujat (tulostustiedot).

Taulukossa 4-14 on kaikki ohjelmayksiköissä käytetyt muuttujat luetteloitu huolimatta niiden käyttötavasta.

Taulukko 4-14 jakaantuu siten, että sen 1-osassa on koko ohjelmiston yhteiset muuttujanimet, 2-osassa on vain kunkin ohjelmayksikön omat muuttujanimet.

Taulukko 4 - 1. Pääohjelman ja muiden ohjelmayksiköiden välinen tiedonsiirto.

FORMALIT ARGUMENTIT		COMMON-MUUTTUJAT			ULKOISET VIITTEET	SISÄISET VIITTEET	DEFINE- PROSEDUURIT
NIMI	I/O	ALUE	NIMI	I/O			
		OS201	NREIT	0	INKONF	YLITYS	
			NURA	0	LUKU		
			NOMI	0	ORJAK		
			NSKAIS	0	PAIVIT		
			NUKAIS	0	POISTU		
			NAUTO	0	URVAIH		
		OS202	XMIN	0	GENERO		
			TKELL	0	LVALOT		
			C	0	JONOT		
			LKMAUT	0	KIIHTY		
			IT	0	KIRJA4		
			XVAP	0			
		OS203	OHJAUS	I			
		OS204	XSEIS	0			
			XLEV	0			
			XVALO	0			
			AALK	0			
			XKAANT	0			
			VKAANT	0			
		OS207	KIRTUL	0			
			MIELKF	0			
			APIEN	0			
			ATSUUR	0			
		OS209	NKONFL	0			

Taulukko 4 - 2. Aliohjelman LUKU ja muiden ohjelmayksiköiden välinen tiedonsiirto.

FORMALIT ARGUMENTIT		COMMON-MUUTTUJAT			ULKOISET VIITTEET	SISÄISET VIITTEET	DEFINE- PROSEDUURIT
NIMI	I/O	ALUE	NIMI	I/O			
URAT	0	OS201	NREIT	I			
PROS	0		NURA	I			
XURA	0		NSKAIS	I			
VURA	0		NUKAIS	I			
VAIHE	0	OS202	C	I			
TILA	0	OS203	OHJAUS	0			
SURA	0		SKALA	0			
EURA	0		TNJAL	0			
IVALO	0						
LV	0						
EPIT	0						
LAMBDA	0						
KIRINP	I						

Taulukko 4 - 3. Aliohjelman INKONF ja muiden ohjelmayksiköiden välinen tiedonsiirto.

FORMALIT ARGUMENTIT		COMMON-MUUTTUJAT			ULKOISET VIITTEET	SISAISET VIITTEET	DEFINE- PROSEDUURIT
NIMI	I/O	ALUE	NIMI	I/O			
		OS206	SEKTOR	0			
KONFLP	0						
KONURA	0						
NURA	0						
NKONFL	I						
NURA2	I						
KIRINP	I						

Taulukko 4 - 4. Aliohjelman PAIVIT ja muiden ohjelmayksiköiden välinen tiedonsiirto.

FORMALIT ARGUMENTIT		COMMON-MUUTTUJAT			ULKOISET VIITTEET	SISAISET VIITTEET	DEFINE- PROSEDUURIT
NIMI	I/O	ALUE	NIMI	I/O			
X	I+0	OS201	NREIT	I			ETAIS
V	I+0		NURA	I			
A	I+0		NSKAIS	I			
SAUTO	I		NUKAIS	I			
EAUTO	I		NAUTO	I			
EAURA	I	OS202	XMIN	I			
VAURA	I		TKELL	I+0			
SURA	I		C	I			
MERKKI	I		IT	I			
TPYS	I+0						
XPYS	I						
AUOMI	I						
XURA	I						
EURA	I						
XVALI	I+0						
URAT	I						
HUOMAU	I						
PITUUS	I						

Taulukko 4 - 7. Aliohjelman GENERO ja muiden ohjelmayksiköiden välinen tiedonsiirto.

FORMALIT ARGUMENTIT		COMMON-MUUTTUJAT			ULKOISET VIITTEET	SISÄISET VIITTEET	DEFINE- PROSEDUURIT
NIMI	I/O	ALUE	NIMI	I/O			
X	I +0	OS101	NREIT	I	EXPJA2 TASAN ORJAK		
V	0		NURA	I			
A	0		NOMI	I			
AU	0		NSKAIS	I			
AUOMI	0		NAUTO	I			
VAURA	0	OS202	TKELL	I			
NRGEN	I +0		C	I			
URAT	I		LKMAUT	I +0			
XURA	I	OS205	IT	I			
LAMBDA	I		LUOKKA	I +0			
TGEN	I +0						
XVALI	0						
EAURA	I +0						
SAUTO	I +0						
EAUTO	I +0						
PROS	I						
HUOMAU	0						
NRPOIS	I						
EPIT	I						
EURA	I						

Taulukko 4 - 8. Aliohjelman LVALOT ja muiden ohjelmayksiköiden välinen tiedonsiirto.

FORMALIT ARGUMENTIT		COMMON-MUUTTUJAT			ULKOISET VIITTEET	SISÄISET VIITTEET	DEFINE- PROSEDUURIT
NIMI	I/O	ALUE	NIMI	I/O			
LVALO	0	OS203	TILA 1	0			
C	I						
TILA	I +0						
VAIHE	I						
LV	I						
IVALO	I						
NSKAIS	I						

Taulukko 4 - 9. Aliohjelman JONOT ja muiden ohjelmayksiköiden
välinen tiedonsiirto.

FORMALIT ARGUMENTIT		COMMON-MUUTTUJAT			ULKOISET VIITTEET	SISAISET VIITTEET	DEFINE- PROSEDUURIT
NIMI	I/O	ALUE	NIMI	I/O			
X	I	OS201	NREIT		KIRJA3		
V	I		NURA				
EAURA	I		NOMI				
JONO	I +0		NSKAIS				
JONOKA	I +0		NUKAIS				
SAUTO	I	OS202	NAUTO				
			XMIN				
		OS204	TKELL				
			XSEIS				
			XLEV				
			XVALO				

Taulukko 4 - 10. Aliohjelman KIIHTY ja muiden ohjelmayksiköiden
välinen tiedonsiirto.

FORMALIT ARGUMENTIT		COMMON-MUUTTUJAT			ULKOISET VIITTEET	SISAISET VIITTEET	DEFINE- PROSEDUURIT
NIMI	I/O	ALUE	NIMI	I/O			
X	I	OS201	NREIT	I	KONFLI ORJAK TASAN	VIHKEL PUNKEL PUNLAH KAANTY	KIIHTI KIIHT2 KIIHT3
V	I		NURA	I			
A	I +0		NOMI	I			
ALFA	I		NSKAIS	I			
EAUTO	I		NUKAIS	I			
SAUTO	I	OS202	NAUTO	I			
VAURA	I		XMIN	I			
EAURA	I		TKELL	I			
XPYS	O		C	I			
AU	I		LKMAUT	I			
AUOMI	I +0	OS204	IT	I			
SURA	I		XVAP	I			
LVALO	I		XSEIS	I			
XURA	I		XLEV	I			
XVALI	I		XVALO	I			
TNKELT	I		AALK	I			
TNVALI	I		XKAANT	I			
EURA	I		VKAANT	I			
KONFLP	I						
KONURA	I						
NKONFL	I						
NKON2	I						
URAT	I						
OHJAUS	I						

Taulukko 4 - 11. Aliohjelman KONFLI ja muiden ohjelmayksiköiden välinen tiedonsiirto.

FORMALIT ARGUMENTIT		COMMON-MUUTTUJAT			ULKOISET VIITTEET	SISÄISET VIITTEET	DEFINE- PROSEDUURIT
NIMI	I/O	ALUE	NIMI	I/O			
X	I	OS201	NREIT	I	ORJAK TASAN KIRJA2	HYVVAL XTOD JALKNF	VARV XARV KIIHT3
V	I		NURA	I			
A	I(+0)		NOMI	I			
AU	I		NSKAIS	I			
AUOMI	I(+0)		NUKAIS	I			
SURA	I		TKELL	I			
EURA	I		C	I			
XURA	I		IT	I			
URAT	I		XVAP	I			
SAUTO	I	OS206	SEKTOR	I			

Taulukko 4 - 13. Aliohjelman SATLUK ja muiden ohjelmayksiköiden välinen tiedonsiirto.

FORMALIT ARGUMENTIT		COMMON-MUUTTUJAT			ULKOISET VIITTEET	SISÄISET VIITTEET	DEFINE- PROSEDUURIT
NIMI	I/O	ALUE	NIMI	I/O			
DIST	'dummy'				RANDOM	TASAN EXPJA1 ORJAK POISSN EXPJA2	

Taulukko 4 - 12. Aliohjelman KIRJA ja muun ohjelmiston välinen tiedonsiirto.

FORMALIT ARGUMENTIT		COMMON-MUUTTUJAT			ULKOISET VIITTEET	SISÄISET VIITTEET	DEFINE- PROSEDUURIT
NIMI	I/O	ALUE	NIMI	I/O			
		OS207	KIRTUL	I			
			MIELKF	I			
			A1PIEN	I			
			A1SUUR	I			
		OS208	KNFPIS	O			
		OS209	NKONFL	I			
<u>ENTRY KIRJA2</u>							
N	I						
X	I						
V	I						
A1	I						
A2	I						
KNFTAL	I						
NAUTO	I						
TKELL	I						
KAPU	I						
XTOD1	I						
XTOD2	I						
AUOMI	I						
<u>ENTRY KIRJA3</u>							
JONMAX	I						
IK	I						
<u>ENTRY KIRJA4</u>							
§	O						
TKELL	I						
<u>SUBROUTINE TILAST</u>							
N	I						
KAPU	I						
LK1	I						
LK2	I						
KONF	I						
XTOD1	I						
XTOD2	I						
V1	I						
V2	I						
A1	I						
A2	I						
TKELL	I						
<u>SUBROUTINE FREKVE</u>							
ARAJA	I						
BRAJA	I						
X	I						
TAULU	O						
N	I						

Taulukko 4 - 14 (osa 1). Koko ohjelmiston yhteiset muuttujanimet.

TIETO- NIMI	MUO- TO	DIMENSIO	KUVAUS
A	R	(NAUTO, IT)	AUTON KIIHTYVYYSMÄTRIIISI, IT-ON VIIIVE AIKA-ASKELEINA
ALFA	R	(NAUTO)	CAR-FOLLOWING-YHTALON HEIKKYYSPARAMETRI
AU	R	(NAUTO, NOM)	AUTON OMINAISUUSMÄTRIIISI (PYSYVÄT OMINAISUUDET)
AU(NAUTO, 1)			AUTON GENEROINTIAJANKOHTA
AU(NAUTO, 2)			AUTON TAVOITENOPEUS
AU(NAUTO, 3)			AUTON HALUTTU JÄRRUTUSHIDASTUVUUS
AU(NAUTO, 4)			AUTON MAKSIMI KIIHTYVYYS
AU(NAUTO, 5)			AUTON EFEKTIIVINEN PITUUS
AUOMI	I	(NAUTO)	AUTON VAIHTUVAT OMINAISUUDET (PAKATTU 1 SANAAN)
AUOMI	BITIT	0,2	KAANTYMINEN =1, SUORAAN =0
AUOMI	BITIT	2,2	KELTAISTA LAHESTYTTÄESSÄ: MENEE =2 PYSÄHTYÄ =1 EI PÄÄTÖSTÄ =0
AUOMI	BITIT	4,1	VARALLA
AUOMI	BITIT	5,4	SOVELLETTAVAN YHTALON MERKKI
AUOMI	BITIT	9,1	POISTUNUT =1, MUUTEN =0
AUOMI	BITIT	10,6	REITTI
AUOMI	BITIT	16,6	NYKYINEN AJOURA
AUOMI	BITIT	22,1	KONFLIKTI =1, EI KONFLIKTIA =0
AUOMI	BITIT	25,1	AJONEUVON TYYPPILUOKKA
C	R		AIKA-ASKELIN PITUUS SEKUNTEINA
EAURA	I	(NURA)	ENSIMMÄINEN AUTO URALLA
FAUTO	I	(NAUTO)	EDellä OLEVAN AUTON IDENTIFIOINTINUMERO
EURA	I	(NURA)	URAN EDESSÄ OLEVAN URIN IDENTIFIOINTINUMEROT
EURA	BITIT	0,6	URAN EDESSÄ SUORAAN JATKUVAN URAN ID-NUMERO
EURA	BITIT	0,6	URAN EDESSÄ KÄÄNTYEN JATKUVAN URAN ID-NUMERO
HUOMAU	I	(NAUTO)	AUTON TILAA KOSKEVA HUOMAUTUS MERKKIJONONA
IT	I		REAKTIOAIKA AIKA-ASKELFINA + 1
IVALO	I		SISAANTULOKAISTOJEN JA JALANKULKUKAISTOJEN SUMMA
KONFLP	I	(NKONFL)	KONFLIKTIPISTEIDEN TIEDOT
KONFLP	BITIT	0,6	LEIKKAAVA AJOURA1
KONFLP	BITIT	0,6	LEIKKAAVA AJOURA2
KONFLP	BITIT	12,7	KONFLIKTIPISTEIDEN ETAISYYS URAN1 ALUSTA
KONFLP	BITIT	19,7	KONFLIKTIPISTEIDEN ETAISYYS URAN2 ALUSTA
KONFLP	BITIT	26,3	SEKTORI, JOSSA URA2 KOHTAA URAN1
KONFLP	BITIT	29,3	SEKTORI, JOSSA URA2 KOHTAA URAN2
KONFLP	BITIT	32,4	PIMEAN KULMAN ETAISYYS KONFLIKTIPISTEESTÄ
KONURA	I	(NKON2)	URALLA OLVAT KONFLIKTIPISTEET
LKMAUT	I		RISTEYKSEEN GENEROITUJEN AUTOJEN LUKUMAARA
LV	I		LIIKENNEVALOVAIHEIDEN LUKUMAARA
LVALO	I	(IVALO)	LIIKENNEVALOJEN NÄYTTÄMÄ: VIHREA =1 VIH-KELTAINEN =2 PUNAINEN =3 PUN-KELTAINEN =4
NAUTO	I		AUTOJEN MAKSIMILUKUMAARA KÄSITTELYSSÄ
NKONFL	I		KONFLIKTIPISTEIDEN LUKUMAARA
NOMI	I		OMINAISUUKSIEN LUKUMAARA MÄTRIIISISSÄ AU
NREIT	I		REITTIEN LUKUMAARA
NRGEN	I	(NSKAIS)	VIIMEIKSI GENEROIDUN AUTON NUMERO SISAANTULOURITTAIN
NRPOIS	I	(NUKAIS)	RISTEYKSESTÄ POISTETUT AUTOT ULOSMENOKAISTOITTAIN
NSKAIS	I		SISAANTULOKAISTOJEN LUKUMAARA
NUKAIS	I		ULOSMENOKAISTOJEN LUKUMAARA
NURA	I		URIN LUKUMAARA
OHJAUS	L		LIIKENNEVALO-OHJAUS TOIMINNASSA =,TRUE,
PROS	R	(NREIT)	LIIKENNEVIHRAN JAKAUTUMINEN SISAANTULOURILTA
SAUTO	I	(NAUTO)	AUTOA SEURAAVAN AUTON IDENTIFIOINTINUMERO
SURA	I	(NURA)	URAA SEURAAVIEN URIN IDENTIFIOINTINUMEROT
SURA	BITIT	0,6	URAA SUORAAN SEURAAVAN URAN ID-NUMERO
SURA	BITIT	0,6	URAA KÄÄNTYEN SEURAAVAN URAN ID-NUMERO
TGEN	R	(NSKAIS)	VIIMEIKSI GENEROIDUN AUTON GENEROINTIAIKA URITTAIN
TILA	R	(2, IVALO)	LIIKENNEVALOJEN LAATU JA KESTO AIKA-ASKELEINA
TKELL	R		SIMULOINTIAJAN KELLO
TNKELT	R	(20)	TODENNAKOISYYSJAKAUTUMA LAHESTYTTÄESSÄ VIH-KELTAISTA
TNVALI	R	(20)	TODENNAKOISYYSJAKAUTUMA AIKAVALIN HYVÄKSYMISELLE
TPYS	R	(NAUTO)	SFISUNTA-AIKA AUTOILLE
URAT	I	(NREIT)	URALUEITTELO REITEITTAIN
V	R	(NAUTO)	AUTON NOPEUS
VAIHE	I	(IVALO, LV)	LIIKENNEVALOJEN KESTOAJAT VAIHEITTAIN
VAURA	I	(NURA)	VIIMEINEN AUTO URALLA
VURA	R	(NURA)	URAKOHTAISET KESKIMÄÄRAISET NOPEUDET
X	R	(NAUTO)	AUTON SIJAINTI KO URAN ALUSTA LUKIEN
XMIN	R		KÄHDEN PERAKKAISEN AUTON MINIMI VALIMATKA
XPYS	R	(NAUTO)	AUTON PYSÄHTYMISPAIKKA
XURA	R	(NURA)	URIN PITUUDET
XVALI	R	(NAUTO)	VALIMATKA EDellä AJAVALAN AUTOON
XVAP	R		VAPAAN ETENEMISEN RAJAMATKA

Taulukko 4 - 14 (osa 2). Ohjelmayksiköiden omat muuttujanimet.

TIETO- NIMI	MUO- TO	DIMENSIO	KUVAUS
PAAOHJELMAN OMAT MUUTTUJAT			
I	I		APUMUUTTUJA
IK	I		SIMULOINTIKIERROSLASKIN
J	I		APUMUUTTUJA
ALIOHJELMAN LUKU OMAT MUUTTUJAT			
I	I		APUMUUTTUJA
IAPU	I	(6)	APUMUUTTUJA REITITITIE TOJEN LUKEMISESSA
IND	I		APUMUUTTUJA KETJUN EURA MUODOSTAMISESSA
ISUM	I		APUMUUTTUJA VALOVAIHEDATOJEN KIERTOAJAN TARKISTUKSESSA
J	I		APUMUUTTUJA
K	I		APUMUUTTUJA
KIERTO	I		LIIKENNEVALOJEN KIERTOAIKA
SUM	R		APUMUUTTUJA REITIDATOJEN TARKISTUKSESSA
TNJAL	R	(4,20)	JALANKULKIJATODENNAKOISYYSJAKAUTUMAT
SKALA	R	(8,2)	TNJAL JA VALOJEN KESTOAIKOJEN VALISET SKAALAUSTEKIJAT
ALIOHJELMAN INKONFL OMAT MUUTTUJAT			
I	I		KASITELTAVANA OLEVAN KONFLIKTIPISTEEN INDEKSI
IA	I		KASITELTAVANA OLEVAN URAN INDEKSI
IAPU	I		APUMUUTTUJA KO KONFLIKTIPISTEEN TALLETTAMISEKSI
IAPV	I		APUMUUTTUJA URALUETTELOJEN KASITTELYSSA
IATALL	I		APUMUUTTUJA KO URAN INDEKSIIN TALLETTAMISEKSI
IB	I		KO KONFLIKTIPISTEEN HISTEAVA URA
IC	I		KONFLIKTIPISTEEN ETAISYYS KO URAA PITKIN METREINA
ID	I		KONFLIKTIPISTEEN ETAISYYS RISTFAVAA URAA PITKIN
IE	I		RISTEAVIEN URIEN VALINEN SUUNTAKULMA
IF	I		NAKOSTEEN ETAISYYS KONFLIKTIPISTEESTA
IND	I		APUMUUTTUJA URIEN KASITTELYSSA
ITESTI	I		KONFLIKTIPISTEDATOJEN LUKUMAARAN TARKISTUSMUUTTUJA
J	I		APUMUUTTUJA
K	I		APUMUUTTUJA OIKEAN URAN ETSIMISESSA
LISA	I		JATKOKONFLIKTIPISTELUETTELOJEN INDEKSI
ALIOHJELMAN PAIVIT OMAT MUUTTUJAT			
ATALL	R	(100)	NYKYISEN KIIHTYVYYDEN TALLETTAUSLASKUKKO
VTALL	R	(100)	NYKYISEN NOPEUDEN TALLETTAUSLASKUKKO
XTALL	R	(100)	NYKYISEN ASEMAN TALLETTAUSLASKUKKO
I	I		KASITELTAVANA OLEVAN URAN INDEKSI
IK	I		KASITELTAVANA OLEVAN ULOSHENOKAISTAN INDEKSI
IKAAANT	I		KAANTYVAN URAN INDEKSI
ISUOR	I		SUORAAN KULKEVAN URAN INDEKSI
J	I		APUMUUTTUJA
KAPU	I		KO AUTON EDESSA OLEVA AUTO
KESKEN	L		LOGGINEN MUUTTUJA URIEN ISUOR JA IKAANT KASITTELYSSA
N	I		KASITELTAVAN AUTON INDEKSI
ALIOHJELMAN POISTU OMAT MUUTTUJAT			
I	I		APUMUUTTUJA
IAPU	I		APUMUUTTUJA
IK	I		APUMUUTTUJA
J	I		APUMUUTTUJA
K	I		APUMUUTTUJA
KAPU	I		AUTON ID=K0001
KAPU1	I		SEURAAJA=AUTON ID=K0001
LKMPOI	I		POISTETTUVEN AJONEUVONJEN LUKUMAARA
NRREIT	I		APUMUUTTUJA REITIN KASITTELYSSA
NRURA	I		APUMUUTTUJA AJONEUVON KASITTELYSSA
XURTOT	R		AJETTU MATKA
VKA	R		KESKIM. MATKANOEPUUS
TMATK	R		MATKA-AIKA
ALIOHJELMAN URVAIH OMAT MUUTTUJAT			
I	I		KASITELTAVAN URAN INDEKSI
IAPU	I		APUMUUTTUJA URJA KASITELTAESSA
IK	I		KIERROSLASKIN
J	I		APUMUUTTUJA
KAPU1	I		APUMUUTTUJA AUTOJA KASITELTAESSA
KAPU2	I		APUMUUTTUJA AUTOJA KASITELTAESSA
N	I		KASITELTAVAN AUTON INDEKSI
NRREIT	I		KASITELTAVAN REITIN INDEKSI
NRURA	I		KASITELTAVAN URAN INDEKSI
ALIOHJELMAN GENERO OMAT MUUTTUJAT			
IRIVI	I		APUMUUTTUJA OIKEAN REITIN ETSIMISESSA
I	I		APUMUUTTUJA
N	I		VIIMEIKSI GENEROITU, MUTTA SIJOITTAMATON AUTO
KAPU	I		VIIMEIKSI SIJOITETTU AUTO
NHAAR	I		HAARAUTUMISIEN LUKUMAARA TIETYSTA SISAANTULOURASTA
NRREIT	I		KASITELTAVAN REITIN APUMUUTTUJA
NRURA	I		KASITELTAVAN URAN APUMUUTTUJA
NRUR1	I		KASITELTAVAN URAN APUMUUTTUJA
SLUKU	I		GENEROITU SATUNNAISLUKU
T	R		AIKAVALI AUTON GENEROIMISESTA SEURAAVAAN HETKEEN
VAGEN	I		URALLE VIIMEIKSI SIJOITETUN AUTON NUMERO
ALIOHJELMAN LVALOT OMAT MUUTTUJAT			
I	I		APUMUUTTUJA KIERROSLASKIMENA
IAPU	I		APUMUUTTUJA
IUDET	I		SEURAAVA VALOJENVAIHTAMISAJANKONTA
TFSTI	L		TESTIMUUTTUJA
TILA1	I	(2,8)	JALANKULKUKAISTOJEN VALOJEN LAATU JA KESTO

jatkuu

Taulukko 4 - 14 (osa 2). Ohjelmayksiköiden omat muuttujanimet.(jatkoa)

TIETO- NIMI	MUO- TO	DIMENSIO	KUVAUS
ALIOHJELMAN JONOT OMAT MUUTTUJAT			
I	I		APUMUUTTUJA
IK	I		KIERROSLASKIN
J	I		APUMUUTTUJA
JPITUU	I		APUMUUTTUJA JONOA KASITELTAESSA
KAPU	I		APUMUUTTUJA AUTOA KASITELTAESSA
ALIOHJELMAN KIIHTY OMAT MUUTTUJAT			
I	I		KASITELTAVANA OLEVAN URAN INDEKSI
IK	I		KASITELTAVANA OLEVAN ULOSMENOKAISTAN INDEKSI
IKAAANT	I		KAANTYVAN URAN INDEKSI
IS	I		TULOSTUSMUUTTUJA
ISUOR	I		SUOHAAN KULKEVAN URAN INDEKSI
ITCH	I		SWITCHI
J	I		APUMUUTTUJA
KAPU	I		EDELLÄ OLEVAN AUTON INDEKSI
KESKEN	L		LOOGINEN MUUTTUJA URIEN ISUOR JA IKAANT KASITTELYSSA
KNFAL	I		KO AUTON EDESSÄ OLEVA KONFLIKTIPISTE
KONFLI	L		LOOGINEN FUNKTIOALIOHJELMA
LKYAUT	I		AUTOJEN LUKUMAARA
N	I		KO AUTON INDEKSI
NRREIT	I		KO REITIN INDEKSI
VKAANT	R		KAANTYVALLA URALLA KESKINOPEUS
XKAANT	R		KAANTYMISLOGIIKAN SOVELTAMISEN ALKUKOHTA
ALIFUNKTION KONFLI OMAT MUUTTUJAT			
ALFA	R		APUMUUTTUJA RISTEAVAN AUTON TULOKULMAN LASKEMISESSA
HAVAIN	I	(NAUTO)	HAVAITUN AUTON INDEKSI
HAVKUL	R		HAVAINNONTOKOKULMA
HYVVAL	L		AIKAVALIN HYVAKSYMISEN LOOGINEN FUNKTIO
I	I		APUMUUTTUJA
IAPU	I		APUMUUTTUJA
IND	I		APUMUUTTUJA
IR	I		APUMUUTTUJA
IREIT	I		APUMUUTTUJA KEITIN KASITTELYSSA
IRIST	I		RISTEAVAN URAN INDEKSI
ISEKTO	I		RISTEAVAN URAN LAHESTYMISSSEKTORI
IURA	I		APUMUUTTUJA URAN KASITTELYSSA
J	I		APUMUUTTUJA
JKNFU	I		KO JALANKULKUSUOJATTIEN INDEKSI
JUTLUK	I		HAVAITUN AUTON INDEKSI (JOKO 1 TAI 2)
K	I		APUMUUTTUJA OIKEAN URAN VALITSEMISESSA
KAPU	I		RISTEAVAA URAA ENSIMMAISENA ETENEVAN AUTON INDEKSI
KONLUK	I		TARKASTELTAVAN KONFLIKTIPISTEEN NUMERO (1,2,...)
L	I		APUMUUTTUJA TODENNAKOISYYSJAKAUTUMA ARPOMISESSA
M	I		KO URAN INDEKSI
MYALL	I		KO URAN INDEKSIIN TALLETUS
TARVI	R		AUTON 1 ARVIOIMA OMA AJOAIKA KONFLIKTIPISTEeseen
TARV2	R		AUTON 1 ARVIOIMA AUTON 2 AJOAIKA KONFLIKTIPISTEeseen
THMAV	R		HAVAITSEMISTODENNAKOISYYS
VIIM	L		URAN VIIMEISEN K-PISTEEN TAPAUKSESSA =,TRUE,
XLISA	R		K-PISTEEN DIMENSIO
XKDNFL	R		K-PISTEEN ETAISYYS URAN ALUSTA
XTOD	R		FUNKTIO AUTON JA K-PISTEEN VALISEN ET, LASKEMISEKSI
XTOD1	R		AUTON 1 ETAISYYS K-PISTEeseen
XTOD2	R		AUTON 2 ETAISYYS K-PISTEeseen
ALIOHJELMAN KIRJA OMAT MUUTTUJAT			
AR	R		APUMUUTTUJA HIDASTUVUUKSIEN TAULUKOINNISSA
EKNFN	R		KESKIMAARAINEN KONFLIKTIPISTE
EKNFV	R		KESKIMAARAINEN KONFLIKTINOPEUS
I	I		APUMUUTTUJA
IA	I		APUMUUTTUJA INDEKSOINNISSA
IB	I		APUMUUTTUJA INDEKSOINNISSA
ISN	I		APUMUUTTUJA INDEKSOINNISSA
J	I		APUMUUTTUJA INDEKSOINNISSA
JONLKM	I	(25)	APUMUUTTUJA JONOPITUUDEN TILASTOINNISSA
JONO	I	(25)	APUMUUTTUJA JONOPITUUDEN TILASTOINNISSA
JRAJA	I		APUMUUTTUJA
KALLE	I	(100,4)	KONFLIKTI TILANTEEN TALLETUS
KERRAT	I		APUMUUTTUJA TULOSTUKSESSA
KNFAPU	I	(25)	APUMUUTTUJA TULOSTUKSESSA
KNFHID	I	(10,100)	KONFLIKTIHIDASTUVUUKSIEN TILASTOINTI
KNFEL	R	(100)	KONFLIKTIEN KESKIM. ESIINTYMISSAJANHETKET
KNFES	I	(6,6910,2)	KONFLIKTIEN TILASTOINTI AJONEUVOLUOKITTAIN
KNFLKM	I	(100)	KONFLIKTIEN LUKUMAARA
KNFNOP	R	(100)	KONFLIKTIEN NOPEUKSIEN TILASTOINTI
KNFTOT	I	(100)	KONFLIKTIEN LUKUMAARAN TILASTOINTI
KNFVAL	R	(100)	KONFLIKTIEN TAPAHTUMISAIKAVALIEN TILASTOINTI
LK1	I		APUMUUTTUJA
LK2	I		APUMUUTTUJA
MRAJA	I		APUMUUTTUJA
NKNFA	I		APUMUUTTUJA
TALLE	R	(100,7)	KONFLIKTITILANTEEN TALLETUS
VALI	R		APUMUUTTUJA FREKVENSIN LASKEMISESSA
ALIFUNKTION SATLUK OMAT MUUTTUJAT			
EXI	R	(2)	KAKSIHUIPPUISEN EXPONENTTIIJAKAUTUJAN KESKIARVOT
I	I		APUMUUTTUJA
M	I		APUMUUTTUJA
R1	R		TASAN JAKAUTUNUT SATUNNAISLUKU
SUM	R		APUMUUTTUJA NORMAALIJAKAUTUMAN GENEROINNISSA
THIN	R	(2)	KAKSIHUIPPUISEN EXPONENTTIIJAKAUTUMAN KESKIARVOMEIOT
TN	R	(2)	KAKSIHUIPPUISEN, EXPONENTTIIJAKAUTUMAN OSUUDET
TR	R		APUMUUTTUJA POISSONIN JAKAUTUMAN GENEROINNISSA
X	R		APUMUUTTUJA POISSONIN JAKAUTUMAN GENEROINNISSA

4.7 I/O - toiminta

4.7.1 Syöttötietojen kuvaus

Taulukossa 4-15 on esitetty syöttötietojen kuvaus silloin, kun ne ovat kortinkuvina. Syöttötietoja valmistettaessa on otettava huomioon seuraavat asiat:

1. Kokonaislukukenttien datat on tasattava oikealle.
2. Desimaalilukukenttien datojen desimaaliosan on täsmättävä formaatin kanssa.
3. Ajourien identifiointikoodien antaminen siten, että sisään-tulokaistat koodataan luvuilla 1...NSKAIS; ulosmenokaistat koodataan luvuilla 1 + NSKAIS ... NSKAIS + NUKAIS; muut ajourat mielivaltaisessa järjestyksessä luvuilla, jotka ovat \leq NURA.
4. Liikennevalojen identifiointikoodi on sama kuin vastaavan sisääntulokaistan id-koodi.
5. Reittien ja konfliktipisteiden identifiointikoodien antaminen tapahtuu mielivaltaisessa järjestyksessä luvuilla 1...NREIT ja 1...NKONFL.

4.7.2 Tulostustietojen kuvaus

Simulaattorin antama tulostus muodostuu konfliktitilanteita karakterisoivien suureiden tilastollisista taulukoinneista. Kohdassa 7.4 on esitetty esimerkkiajon tulostus. Osa simuloinnin tulostuksesta ohjataan massamuistin tiedostoihin, jolloin on mahdollista simulointiajan jälkeen suorittaa aineistolle tilastollisia analyysyjä. Suoritettavien analyysien tarkempi määrittäminen on jätetty myöhemmin suoritettavaksi, kun tutkittavia ilmiöitä karakterisoivat suureet on ensin tiedostettu.

Tiedostoon ohjataan myös tietoa, jonka avulla on lisäksi mahdollista piirtää piirturilla ilmavalokuvaa muistuttavia

kuvia risteyksestä esimerkiksi silloin, kun on sattunut vakava konflikti tai onnettomuus. Piirtäminen on toteutettu VTT:n rumpupiirturilla CALCOMP 925/136. "Valokuva-piirroksen" kuten muidenkin tulostustietojen jälkikäsitteily jää tämän työn ulkopuolelle.

Tulostuksen ensimmäinen osa käsittää valinnaisen syöttötietojen listauksen. Tämä osa listataan vain, jos syöttötiedoissa luettu parametri KIRINP saa nollasta poikkeavan arvon.

Konfliktisuuden mittaamiseen käytetään toimintojen "häätäisyksistä" kerättyjä tietoja. Toiminnan hätäisyyden mittaaminen suoritetaan jarrutushidastuvuusarvojen perusteella. Varsinaisen tulostuksen muodostuu simuloinnin kumulatiivisista taulukoinneista: konfliktipistekohtaiset jarrutushidastuvuusjakautumat ilmaisevat kunkin konfliktipisteen suhteellisen vaarallisuuden, konfliktipistekohtaiset tiedot keskimääräisesti konfliktien tapahtumisaikavälistä ja ajoneuvojen nopeuksista ennen konfliktia antavat lisätietoa konfliktianalyysiä varten, ajoneuvoluokkien väliset konfliktijakautumat koko risteyksen aineistosta ja valinnaisesti tietyistä konfliktipisteistä kuvaavat eri ajoneuvotyyppien "vaarallisuutta". Risteyksen toimintaa karakterisoitavista suureista tulostetaan ainoastaan sisääntulokaistojen maksimijononpituudet.

Edellä kuvattu tulostus voidaan suorittaa enintään 10 eri ajanhetkenä simuloinnin kuluessa simuloinnin lopussa suoritettavan tulostuksen lisäksi. Rivikirjoittimella suoritettavan tulostuksen kuvaus on taulukossa 4-16.

Simuloinnin kuluessa tulostetaan kolmeen tiedostoon jatkoanalysointia varten kustakin konfliktitilanteesta yhteenvetotiedot, kustakin systeemissä kulkeneesta ajoneuvosta tietyjä ominaisuuksia ja konfliktitietoa sekä koko risteyksen "koodikuva" tietyissä konfliktitilanteissa. Tiedostoihin talletettavista tiedoista on taulukoissa 4-17...4-19 tarkempi kuvaus.

Simulaattorilla voidaan tulostaa myös koko risteyksen toimintaa

Taulukko 4 - 15. Syöttötiedot.

Tietonimi	Syöttö- laite	Formaatti	Sarakkeet	Sisällön kuvaus ja muut seikat (rajoitukset, oletusarvot)
KIRJOI	1.kortti	F6.0	6 - 11	ajoneuvojen dyn.käyttäytymisen tulostusoptio
TSIMU	2.kortti	F6.0	6 - 11	simulointiaika (sek)
KIRINP	"-	I1	12	oletus: tyhjä ei listaa inputia
KIRTUL(10 alk.)	"-	10I5	13 - 62	välitulostushetket (sek)
MIELKF(9 alk.)	"-	9I2	63 - 80	konfliktipisteet, joista halutaan ajoneuvoluokkien vähenem. jakautuma
MRAJA	3.kortti	I1	1	em. konfliktipisteiden lkm.
APIEN	"-	F4.2	2 - 5	ensimmäiseen ajoneuvoluokkien väliseen jakautumaan laskettavien konfliktien suurin hidastuvuusraja
AISUUR	"-	F4.2	6 - 9	toiseen - " -
TNKELT(20 alk.)	4.kortti	20F4.3	1 - 80	ajajan tn-jakautuma keltaista lähestyttäessä
TNVALI(20 alk.)	5.kortti	20F4.3	1 - 80	aikavälin hyväksyminen tn.-jakautuma
NURA	6.kortti	I3	1 - 3	ajourien lkm.
NSKAIS	"-	"-	4 - 6	sisääntulokaistojen lkm.
NUKAIS	"-	"-	7 - 9	ulosmenokaistojen lkm.
NREIT	"-	"-	10 - 12	ajoreittien lkm.
NKONFL	"-	"-	13 - 15	konfliktipisteiden lkm.
LV	"-	"-	16 - 18	liikennevalovaihteiden lkm. jaksossa
JAL	"-	"-	18 - 21	jalankulkijain suojateiden lkm.
C	7.kortti	F4.2	1 - 4	simuloinnin aika-askel (sek)
XVAP	"-	"-	5 - 8	car-fallowingin ja vap.eten.raja
XMIN	"-	"-	9 - 12	ajoneuvojen minimietäisyys (m)
XSEIS	"-	"-	13 - 16	seis-viivan paikka (m) uran alusta
XVALO	"-	"-	17 - 20	liikennevalon sijainti uran alusta (m)
XLEV	"-	"-	21 - 24	ajokaistan leveys (m)
XKAANT	"-	"-	25 - 28	kääntymisen alkukohta (m)
YKAANT	"-	"-	29 - 32	kääntymisnopeus (m/s)
AALK	"-	"-	33 - 36	alkukiiht.(vapaa etenem.(m/s ²))
SEKTOR	8.kortti	F4.0	6 - 37	konfliktipistelähestymisen sektorijaotus (asteina)

Jatkuu

Taulukko 4 - 15. Syöttötiedot. (jatkoa¹)

Tietonimi	Syöttö- laite	Formaatti	Sarakkeet	Sisällön kuvaus ja muut seikat (rajoitukset, oletusarvot)
ITESTI	9.kortti	I2	6 - 7	testisuure konfliktipisteiden luvussa
FLD(0,6,KONFLP(i))	"-	I2	11 - 12	i:nnen konfliktipisteen 1.ajour
FLD(6,6,KONFLP(i))	"-	I2	13 - 14	- " - 2.ajour
FLD(12,7,KONFLP(i))	"-	I3	16 - 18	i:nnen k-pisteen sijainti 1.ajorua
FLD(19,7,KONFLP(i))	"-	I3	19 - 21	- " - 2.ajorua
FLD(26,3,KONFLP(i))	"-	I4	22 - 25	ajourien kohtauskulma i:nnessä k-pisteessä, jos 2. tulee 1. nähdn oikealta, negatiivinen kulma jos 2. tulee 1. nähdn vasemmal positiivinen kulma
FLD(32,4,KONFLP(i))	"-	I4	26 - 29	"pimeän kulman" etäis.k-pistees jos on = 0, ei ole pimeitä kulmaa
.				
.				
.	NKONFL kpl samanlaisia kortteja kuin edellä oleva			
FLD(0,6,URAT(i))	kortti	I2	11 - 12	1. ajoura reitillä i
FLD(6,6,URAT(i))	"-	"-	13 - 14	2. - " -
FLD(12,6,URAT(i))	"-	"-	15 - 16	3. - " -
FLD(18,6,URAT(i))	"-	"-	17 - 18	4. - " -
FLD(24,6,URAT(i))	"-	"-	19 - 20	5. - " -
FLD(30,6,URAT(i))	"-	"-	21 - 22	6. - " -
PROS(i)	"-	"-	25 - 27	liikennemäärän suhteellinen osu 1.ajorualla generoidusta mää- rystä
.				
.				
.	NREIT kpl samanlaisia kortteja kuin edellä oleva			
IAPU(6)	kortti	I2	6 - 7	testisuure ajourien luvussa
FLD(0,6,SURA(i))	"-	I2	11 - 12	i:nnen ajouran suora seur.ajour
FLD(6,6,SURA(i))	"-	I2	13 - 14	"- käänt.seur.ajour
XURA(i)	"-	F4.1	16 - 19	"- pituus (m)
VURA(i)	"-	F3.1	23 - 25	"- keskim.tav.nopeus (m/s)
LAMBDA(1,i)	"-	F4.1	26 - 29	gener.aikavälin 1. keskiarvo
LAMBDA(2,i)	"-	"-	30 - 33	"- 2. "-
LAMBDA(3,i)	"-	"-	34 - 37	"- minimiaika T ₁
LAMBDA(4,i)	"-	"-	38 - 41	"- "- T ₂

Jatkuu

Taulukko 4 - 15. Syöttötiedot. (jatkoa²)

Tietonimi	Syöttö- laite	Formaatti	Sarakkeet	Sisällön kuvaus ja muut seikat (rajoitukset, oletusarvot)
LAMBDA(5,i)	kortti	F3.2	42 - 44	1.:n suht.osuus generoinnissa
EPIT(1,i)	"	F4.2	45 - 48	ajoneuvolajin 1 suht.osuus
EPIT(2,i)	"	F4.1	49 - 52	"- pituuden keski- arvo (m)
EPIT(3,i)	"	F2.1	53 - 54	"- pituuden hajont
EPIT(4,i)	"	F2.0	55 - 56	"- tyypiluokka (1...5)
EPIT(5,i)	"	F4.2	57 - 60	ajoneuvolajin 2 suht.osuus
EPIT(6,i)	"	F4.1	61 - 64	"- pituuden keski- arvo
EPIT(7,i)	"	F2.1	65 - 66	"- pituuden haonta
EPIT(8,i)	"	F2.0	67 - 68	"- tyypiluokka
EPIT(9,i)	"	F4.2	69 - 72	ajoneuvolajin 3 suht.osuus
EPIT(10,i)	"	F4.1	73 - 76	"- pituuden keski- arvo
EPIT(11,i)	"	F2.1	77 - 78	"- pituuden hajont
EPIT(12,i)	"	F2.0	79 - 80	"- tyypiluokka
NSKAIS kpl samanlaisia kortteja kuin edellä oleva				
IAPU(6)	kortti	I2	6 - 7	testisuure ajourien luvussa
FLD(0,6,SURA(i))	"	I2	11 - 12	i:nneen ajouran suora seur.ajour
FLD(6,6,SURA(i))	"	I2	13 - 14	"- kääntyvä "-
XURA(i)	"	F4.1	16 - 19	"- pituus (m)
VURA(i)	"	F3.1	23 - 25	"- keskim.tavoite- nopeus
NURA-NSKAIS kpl samanlaisia kortteja kuin edellä oleva				
OHJAUS	kortti	L1	1	T, jos valo-ohj.päällä muul- loin F
KIERTO	"	I3	2 - 4	valo-ohjauksen jakson pituus (sek)
TILA(1,i)	*kortti	I1	11	liikennevalon i tila jakson alussa i = 1 vihreä i = 2 vih- kel i = 3 punainen i = 4 pun- kel
TILA(2,i)	"	I2	13 - 14	liikennevalon i tilan kesto jakson alusta lukien

Jatkuu

Taulukko 4 - 15. Syöttötiedot. (jatkoa³⁾)

Tietonimi	Syöttö- laite	Formaatti	Sarakkeet	Sisällön kuvaus ja muut seikat (rajoitukset,oletusarvot)		
VAIHE(1,i)	*kortti	I2	16 - 17	tilan 1 kesto aika	jaksossa	
VAIHE(2,i)	"-	"-	18 - 19	tilan 2	"-	"-
VAIHE(3,i)	"-	"-	20 - 21	tilan 3	"-	"-
VAIHE(4,i)	"-	"-	22 - 23	tilan 4	"-	"-
VAIHE(LV,i)		"-	78 - 79	tilan LV	"-	"- (LV on yleensä = 4)
,						
.						
.	IVALO kpl samanlaisia kortteja kuin edellä oleva					
TNJAL(1,20)	*kortti	20F4.3	1 - 80	jalankulkijan tn-jakautuma lähtien	kadun reunaa	
TNJAL(2,20)	"-	"-	"-	jalankulkijan tn-jakautuma 2.	k-pisteessä	
TNJAL(3,20)	"-	"-	"-	jalankulkijan tn-jakautuma 3.k-	kadun reunasta lukien	
TNJAL(4,20)	"-	"-	"-	jalankulkijan tn-jakautuma 4.	k-pist.kadun reunasta lukien	
NTUTKO	kortti	I2	1 - 2	havaitsemisen kauaskantoisuus-parametri (ajoneuvon edestä tutkittavien konfliktipisteiden lkm.)		

*) liikennevaloja koskevat datat luetaan vain, jos tietonimen OHJAUS arvo on T.

Taulukko 4 - 16. Tulostustiedot.

Tietonimi	Tulostus- väline	Sijainti (otsik- ko,diaar.,ehdoll.)	Formaatti	Muut (laatu, skaalaus, virherajat yms.)
TKELL	rivikirjoitin	TULOSETAAN...	F6.0	(sek)
I	---	KONFLIKTIPISTE	I9	
KNFTOT(1)	---	KONFLIKTIEN LU- KUMÄÄRÄ	I3	(kpl)
EKNFV	---	KESKIM.TAPAHTUMIS.	F13.1	(sek)
EKNFN	---	KESKIM.NOPEUS...	F14.1	(m/s)
KNFHID(1, i)	---	JARRUTUSHIDASTU- VUUS...	I7	(m/s ²)
KNFHID(2, i)	---	---	---	---
KNFHID(3, i)	---	---	---	---
KNFHID(4, i)	---	---	---	---
KNFHID(5, i)	---	---	---	---
KNFHID(6, i)	---	---	---	---
KNFHID(7, i)	---	---	---	---
KNFHID(8, i)	---	---	---	---
KNFHID(9, i)	---	---	---	---
KNFHID(10, i)	---	---	---	---
NKNF	---	konflikteja yht.	I11	(kpl)
KNFAPU(1)	---	konfl.alle-9,0 m/s ²	---	---
KNFAPU(2)	---	--- -8,0m/s ²	---	---
KNFAPU(10)	---	--- -0,0m/s	---	---
AR	---	AJONEUVOLUOKKIEN...	F7.1	rajahidastuvuus, jota vakavammat konfliktit taulukoidaan
MIELKF	---	---	I3	konfliktipisteen, josta taulukoidaan ajoneuvo- luokkien välinen jakautum
KNFKES(i, j, k, l)	---	---	I5	i = käsiteltävänä oleva ajoneuvo j = konfliktoiva ajoneuvo k = konfliktipiste, josta taulukko on kerätty, jos k=1, kyseessä on koko risteys l = taulukkoon kerättävän aineiston vakavuusast yleensä l=1 tai 2
I	---	SISÄÄNTULOKAISTOIL- LA	I4	sisääntulokaistan id-kood
KNFAPU(I)	---	---	I4	edellistä vastaavan jonon maksimipituus (laatu: ajonopeus)

Taulukko 4 - 17. Tiedoston FKNF määrittely ja tietueen kuvaus.

Tiedoston atk-tunnus FKNF		Tiedostoa kuvaava nimi Konfliktitilanteet		
Taltio (t)		Tiedoston koko (merkkiä tms.)		Tietokone UNIVAC 1108
Tiedoston rakenne		Määrittelyt (block, avain tms.)		
Käyttöoikeus vapaa		Tiedostoa käyttäviä projekteja ja ohjelmia		
Tietueen tunnus KONFLI		Kuvaava nimi Konfliktitilanne		Tietueitten lkm. vaihtuva
Kent- än nro	Tieto	Kentän pituus	Esitys- muoto	Tietonimi
1	käsiteltävänä oleva ajoneuvo	data-sana	formatoima- ton (binäärinen)	N
2	konflikoiva ajoneuvo	"-	"-	KAPU
3	käsiteltävänä olevan ajoneuvon luokkatyyppi	"-	"-	LK1
4	konflikoivan ajoneuvon luokka- tyyppi	"-	"-	LK2
5	konfliktipisteen id-koodi	"-	"-	KNFTAL
6	käs.olevan ajoneuvon etäisyys konfliktipisteestä	"-	"-	XTOD1
7	konflikoivan ajoneuvon etäisyys k-pisteestä	"-	"-	XTOD2
8	ko.ajoneuvon nopeus	"-	"-	V1
9	konfl.ajoneuvon nopeus	"-	"-	V2
10	ko.ajoneuvon hidastuvuus	"-	"-	A1
11	konfl. "-	"-	"-	A2
12	konfliktin ajanhetki	"-	"-	TKELL

Taulukko 4 - 18. Tiedoston FAUTO määrittely ja tietueen kuvaus.

Tiedoston atk-tunnus		Tiedostoa kuvaava nimi				
FAUTO		Ajoneuvojen talletus				
Taltio		Tiedoston koko (merkkiä tms.) Tietokone				
Tiedoston rakenne		Määrittelyt				
Käyttöoikeus		Tiedostoa käyttäviä projekteja ja ohjelmia				
Tietueen tunnus		Kuvaava nimi		Tietueitten lkm.		
Ken- tän nro	Tieto		Kentän pituus	Esitys- muoto	Tietonimi	
1	ajoneuvon id-koodi		data-sana	formatoima- ton (binäärinen)	KAPU	
2	"- ajoreitti		"-	"-	NRREIT	
3	ajoneuvon matkalla sattunut 1. konfl.		"-	"-	IAPU(1)	
4	"-	"-	"-	2. "-	"-	IAPU(2)
5	"-	"-	"-	3. "-	"-	IAPU(3)
.	
.	
.	
12	ajoneuvon matkalla sattun. 10. konfl.		"-	"-	IAPU(10)	
13	ajoneuvon keskim. nopeus		"-	"-	VKA	
14	pysähtyneenä oloaika (sek.)		"-	"-	TPYS	

Taulukko 4 - 19. Tiedoston FKUVA määrittely ja tietueen kuvaus.

Tiedoston atk-tunnus FKUVA		Tiedostoa kuvaava nimi Konfliktitilanteiden piirtäminen		
Taltio		Tiedoston koko (merkkiä tms.)		Tietokone UNIVAC 1108
Tiedoston rakenne		Määrittelyt (block, avain tms.)		
Käyttöoikeus		Tiedostoja käyttäviä projekteja ja ohjelmia		
Tietueen tunnus RISTEYS		Kuvaava nimi Risteyksen hetkell.kuva		Tietueitten lkm. vaihtuva
Kent- tunnus	Tieto	Kentän pituus	Esitys- muoto	Tietonimi
1	paikkakoordinaatti	data-sana	formatoima- ton (binäärinen)	X(1)
:	:	:	:	:
NAU	-"	-"	-"	X(NAU)
	ajoneuvojen ominaisuudet	-"	-"	AUOMI(1)
	:	:	:	:
	-"	-"	-"	AUOMI(NAU)
	seuraaaja-ajoneuvo	-"	-"	SAUTO(1)
	:	:	:	:
	-"	-"	-"	SAUTO(NAU)
	ensimmäinen ajoneuvo uralla	-"	-"	EAURA(1)
	:	:	:	:
	-"	-"	-"	EAURA(NURA)
	viimeinen ajoneuvo uralla	-"	-"	VAURA(1)
	:	:	:	:
	-"	-"	-"	VAURA(NURA)

esittävä simuloinnin aika-askelen tarkkuudella jatkuva "koodikuva", mutta sopivan tulostuslaitteen puuttumisen vuoksi tämä osa tulostusta ei ole käytössä, sillä esim. rivikirjoittimella tulostettaessa koodikuva on epähavainnollinen ja ko. tulostus kuluttaisi koneaikaa suunnattomasti.

4.7.3 Virheilmoitukset

Taulukossa 4-20 on lueteltu ohjelman logiikan havaitsemien virhetilanteiden selvittely ja mahdolliset toimenpiteet. Osa virheistä johtaa ajon keskeytykseen (STOP-käsky), osa taas on luonteeltaan diagnostisia (diag), jolloin ajo jatkuu virheestä huolimatta.

Taulukko 4 - 20. Virheilmoitukset.

Ohjelmayksikkö	Virheilmoitus	Selvitys ja toimenpiteet
pääohjelma	XXX:N LUKUMÄÄRÄ LIIAN SUURI	Taulukon dimensioraja liian pieni input-dataan nähden (STOP)
LUKU	REITTIA XXX ENNEN LUETUT LIIKENNEMÄÄRÄDATAT VIRHEELLISIA, SUMMA EI OLLUT = 1 (FRMT 1086)	input-datojen tarkistus (diag)
"-"	URATOJEN LUKUMÄÄRÄ VIRHEELLINEN (FRMT 1093)	input-datojen tarkistus (diag)
"-"	XXX:N VALOVAIHEIDEN SUMMA EI TÄSMÄNNYT KIERTOAJAN KANSSA (FRMT 1097)	input-datojen tarkistus (diag)
"-"	REITTIKOHTAISTEN URALUETTELODATOJEN LUKU EI ONNISTUNUT (FRMT 1098)	input-datojen lkm.-tarkistus (STOP)
"-"	URAN XXXX AJONEUVOLAJIDATOIS- SA VIRHE (FRMT 1062)	input-datojen tarkistus (diag)
INKONF	KONFLIKTIPISTEDATOJEN LUKUMÄÄRÄSSÄ VIRHE (FRMT 1177)	input-datojen tarkistus (diag)
"-"	XXX:NNEN KONFLIKTIPISTEEN SIJOITTAMINEN EI ONNISTUNUT (FRMT 1171)	taulukon KONFLP käsittely epäonnistui (STOP)

jatkuu

Taulukko 4 - 20. Virheilmoitukset (jatkoa 1)

Ohjelmayksikkö	Virheilmoitus	Selvitys ja toimenpiteet
PAIVIT	XXXX:NNEN URAN KÄSITTELYSSÄ VIRHE. AUTON ID-KOODI: XXXX	taulukoiden VAURA ja SAUTO sisältö yhteensopimaton (STOP)
URVAIH	KÄSITELTÄVÄN AUTON URATIETO- EI TÄSMÄÄ AUTON REITIN URATIEDON KANSSA (FRMT 2293)	(diag)
-"-	URALUETTELON MUKAAN SEURAAVA EI VOI OLLA URA XXX (FRMT 2209)	ajoneuvon uusi ura ei ole taulukon EURA mukainen (STOP)
GENERO	KÄSITELTÄVÄN OLEVA URANUMERO- JA AUTON URATIETO ERILAISIA (FRMT 2408)	(diag)
-"-	KAIKKI AUTOLLE VARATUT INDEKSIT OVAT KÄYTÖSSÄ (FRMT 2499). TAULUKON NRPOIS DUMPPAUS	Risteyksessä on maksimimäärä ajoneuvoja. Suurennä dimenssioraja (STOP)
LVALOT	TAULUKOSSA TILA EI-POSITIIVINEN ALKIO (FRMT 2590)	(diag)
JONOT	JONOSTA XX EI LÖYTÄNYTKÄÄN XXXX AUTOA (FRMT 2692)	Jonon pituus on muuttunut jostaisyystä (STOP)
KONFLI	RISTIRIITA TAULUKOIDEN KONURA JA KONFLP TIEDOISSA (FRMT 2892)	Tarkasta ko. taulukot (STOP)
-"-	RISTÄVÄN URAN XXXX LUETTELOSTA EI LÖYTÄNYTKÄÄN KONFLIKTIPISTETTÄ XXXX (FRMT 2893)	Vaikka konfliktipiste on paikallistettu ko. uralle (STOP)
-"-/XTOD	AUTO ON POISSA REITILTÄÄN (FRMT 3490)	XTOD voidaan laskea vain auton reitillä olevaan konfliktipisteeseen (STOP)
-"-/XTOD	RISTIRIITA TAULUKOSSA KONFLP JA URANUMERON VÄLILLÄ (FRMT 3492)	(STOP)
-"-/JALKNF	JALANKULKIJAN LIIKENNEVALOKOODI EI VOI OLLA = XXXX (FRMT 3590)	Ko. koodi voi olla = 1 tai = 3 (STOP)

4.8 Atk-laitteisto ja käyttö

4.8.1 Atk-laitteisto

Tietokoneohjelma on koodattu FORTRAN V:llä ja se on käännetty, ajettu ja testattu UNIVAC 1108-tietokoneella, EXEC 8-käyttöjärjestelmän alaisena. Käyttöjärjestelmäversio ei ole ohjelman ajolle merkitsevä. Sen sijaan ohjelman koneriippuvuusaste on sellainen, että se on helpohkosti konvertoitavissa muille tietokoneille kohdassa 4.4 lueteltujen muutosten mukaan.

Ohjelma vaatii 43 Ksanaa keskusmuistitilaa, josta ohjelman käskyjen osuus I-pankissa on 20 Ksanaa ja käsiteltävien tietojen osuus D-pankissa on 23 Ksanaa. Muistitilan tarve D-pankin osalta riippuu luonnollisesti simuloitavan risteyksen koosta. Em. arvot ovat esimerkkiajosta, jossa simuloitiin 13-sisääntulokaistaista risteystä ja tulostus käsitti kaikki mahdollisen tulostuksen risteyksen dynaamisen tilan tulostus mukaanluettuna. Ohjelman koko mitattuna reikäkortteina on n. 1600 reikäkorttia.

Simuloinnin suorittamisessa tarvittavat oheislaitteet ovat kortinlukija (välttämätön), rivikirjoitin (välttämätön), rumpumuisti tai muu muistiväline(mahdollinen) ja piirturi (mahdollinen) sekä näyttöpääte (löytyisi käyttöä).

4.8.2 Käyttö UNIVAC 1108:ssa

Simulointiohjelman KNFSIM käyttömuoto U-1108:ssa on eräajo. Tietokoneajoaika on suoraan riippuva vapaasti valittavissa olevasta simulointiajasta. Ohjelman taltioidut ovat luonnollisten lista- ja korttitaltioiden lisäksi magneettinauhalla. Ohjelma valmistui 1974-10-08, jonka päivämäärän tasolla nämä dokumentit ovat.

Ohjelman vaatima ajovirta on esitettävissä kuvan 4-24 mukaisesti luettaessa koko ohjelma ja datat korteilta. Sen jälkeen kun ohjelma on talletettu tiedostoon FSIMU, ajovirta muodostuu tiedostojen varauskäskyjen (optiota A käyttäen) lisäksi "XQT ja "ADD-käskyistä".

Syöttölaite Laji

kortinlukija	C	"RUN,	
"-"	C	"PSW	
"-"	C	"ASG, UP FKNF., F2	(varataan tulostustiedost)
"-"	C	"ASG, UP FAUTO., F2	"-"
"-"	C	"ASG, UP FKUVA., F2	"-"
"-"	C	"ASG, UP FSIMU., F2	(varataan ohjelmatiedosto)
"-"	C	"ASG, UP FDATAT., F2	(varataan datatiedosto)
"-"	C	"ELT, IL FDATAT., YKSI	(luetaan datat tiedostoksi)
"-"	D	syöttötiedot	
"-"	C	"END YKSI	
"-"	C	"FOR, IS FSIMU. PAA.,. RPAA	(pääohjelman kääntäminen)
"-"	P	pääohjelma	
"-"	C	"FOR, IS FSIMU. LUKU.,.RLUKU	(aliohjelman LUKU
"-"	P	aliohjelma LUKU	määrittäminen)
"-"	C	"FOR, IS FSIMU. INKONF.,.RINKON	
"-"	P	aliohjelma INKONF	
		.	vastaavalla tavalla muut
		.	aliohjelmat
"-"	C	"MAP, I , FSIMU. ABS	(absoluuttielementin muodostaminen)
"-"	C	"IN FSIMU. RPAA.,.RLUKU.,.RINKON ,...	
"-"	C	"XQT FSIMU. ABS	(toteuttaminen)
"-"	C	"ADD FDATAT.	
"-"	C	"FIN	

Laji: C = ohjauskäsky
P = ohjelma tms. pakollista
D = data (syöttötiedot)

Kuva 4 - 23. Ajovirtakaavio

5. SIMULOINTIMALLIN VALIDISOINTI JA JATKOTUTKIMUKSESTA

5.1 Validisointikriteerejä

Koska simulaattorin rakentamisen tarkoitus on jäljitellä reaali maailman tapahtumia, on hyvin tärkeätä osoittaa simulointimallin toiminta oikeaksi. Tämä mallin validisointi voidaan jakaa teknisen validiteetin, premissivaliditeetin ja kokonaisvaliditeetin tarkasteluksi.

Teknisellä validiteetilla tarkoitetaan sitä, onko mallin koodaus suoritettu oikein sekä muodollisesti että loogisesti mallissa esitettyjen toimintojen mukaisesti. Simulointimallin tekninen validiteettitestaus on suoritettavissa pääosin samoilla menetelmillä kuin yleensäkin tietokoneohjelman testaus: ohjelman toiminnan manuaalinen tarkastelu askel askeleelta ja ohjelman logiikan pöytä- ja konetestaus. Ohjelman logiikan konetestausta vaikeuttaa kuitenkin mallin stokastisuus, sillä loogiset virheet voivat helposti sekaantua satunnaistekijöiden aiheuttamiin vaihteluihin. Tämän vuoksi testausvaiheessa satunnaismuuttujat ensin korvattiin vakioilla, jolloin osamallien toiminta oli helposti laskettavissa ja kontrolloitavissa. Seuraavassa vaiheessa koko mallin toiminnan logiikka testattiin käyttämällä vain yhtä satunnaislukusarjaa, joka on tuotettavissa, kun käytetään vain yhtä siemenlukua.

Premissivalidisoinnissa tarkastellaan malliin tehtyjen oletusten ja relaatioiden oikeellisuutta. Tässä mallin testausvaiheessa suoritettiin ajan suhteen lyhyitä simulointiajoja ja systeemin kunkin ajoneuvon operatiiviset suureet tulostettiin jokaisen simuloinnin aika-askelen päätyttyä. Ajoneuvojen liikkeitä ja tehdyt päätökset analysoitiin ja tutkittiin, vaikuttivatko ne järkeviltä. Tällainen testaus suoritettiin kahdessa vaiheessa: mallin eri komponenttien toimintaa tarkasteltiin kutakin erikseen ja sitten koko mallia yhtenä kokonaisuutena. Tällä tavoin helpottui

mallin eri osien välisten loogisten kytkentöjen muodostaminen malleiksi. Osamalleista erikoisesti pysähtymismalliin ja edeltäjän seuraamismalliin kiinnitettiin huomio, koska niitä koskevia dataa oli saatavissa kirjallisuudesta. Eri parametrien vaikutusta mallin käyttäytymiseen ja erikoisesti konfliktisuuteen tutkittiin herkkyysanalyysillä. Kohdassa 3.6.3 esitetyt parametrit: kuljettajan päätöksenteon (havainnointeja) kauaskantoisuus ja reaktioaika osoittautuivat voimakkaimmin konfliktien määrään vaikuttavaksi. Muiden parametrien vaikutuksen tarkempi analysointi jätettiin suoritettavaksi vasta sitten, kun malli on kalibroitu johonkin todelliseen risteykseen.

Premissi- ja kokonaisvalidisoinnin suorittamisen suurimpana vaikeutena on tässä vaiheessa se, että risteyksen toiminnasta ei ollut saatavissa eksakteja dataa. Tarkemman suorittamisen edellytyksenä olisi kenttätutkimuksissa mitattujen liikenteen karakterististen suureiden olemassaolo, jolloin niitä voitaisiin verrata kehitetyn mallin tuottamiin dataihin. Vaikka simuloinnin tarkoituksena ei ole jäljitellä täydellisesti todellisuutta kaikissa sen yksityiskohdissaan, simulointitulosten on oltava "riittävästi" yhteensopivia reaali maailman kanssa, jotta mallia voitaisiin käyttää suunnittelutarkoituksiin.

Simulaattorin kokonaisvalidisointikriteerien tulisi perustua suunnittelutarkoituksiin käytettyihin operatiivisiin kriteereihin. Koska risteyssysteemin tietyn aspektin mukainen (esim. konfliktisuus) suunnittelukriteeri voi olla toinen kuin toisen aspektin mukainen (esim. liikennevalojen ajoitus) suunnittelukriteeri, ei tällaiselle yleiselle simulaattorille ole aina-käypää toimintakriteeriä.

Risteyksiliikenteen simulointimallin toiminnan validiteettikriteereinä voidaan tarkastella ainakin seuraavia suureita:

1. konfliktin määrä ja vakavuusaste
2. risteyksen läpikulkuaika

3. kunkin liikennevirran keskimääräinen tiheys ja nopeus
4. pysähtymisten lukumäärä
5. jonojen pituudet eri paikoissa
6. välietäisyyksien, nopeuksien ja kiihtyvyyksien jakautumat tietyissä tarkkailukohteissa
7. konfliktikäyttäytyminen konfliktoivien ajoneuvojen osalta ja konfliktin vaikutus muuhun risteysliikenteeseen.
8. ajoneuvotyyppien ominaisuuksien jakautumat.

Suoritettaessa validisointia em. kriteerien pohjalta on välttämätöntä saada vastaavat validisointidatat joko kenttätutkimuksista tai liikennevirtatutkimusta käsittelevästä kirjallisuudesta. Tämän työn toimeksiantoon ei sisällytetty ko. jakautumadatojen tarkastelua. Koska kenttätutkimuksilla saatavien datojen keräämistä useinkin rajoittavat sekä taloudelliset että käytännölliset tekijät, on monesti tyydyttävä mallin oikeellisuutta osoitettaessa toteamaan tutkittavan suureen käyttäytyvän mallissa ja todellisuudessa kvalitatiivisesti samalla tavalla.

5.2

Simulointiajojen suorittamisesta

Aivan samoin kuin todellisuudessa luotettavan tuloksen aikaansaamiseksi kenttämittauksia on suoritettava lukuisia kertoja, stokastisessa simuloinnissa tietokoneajoja on suoritettava riittävän monta kertaa.

Simulointimallin hyväksikäytössä ja varsinaisten simulointiajojen suorituksessa keskeisimmiksi piirteiksi tulevat statistiset ongelmat. Varsinaiset simulointiajothan edustavat satunnaisesti valittuja reaali maailman otoksia, jolloin heti joudutaan kysymään, kuinka monta tällaista satunnaisotosta tarvitaan.

Koska Monte Carlo-simulointi on luonteeltaan stokastista,

simuloinnin tuloksissa esiintyy tilastollista vaihtelua. Täten Monte Carlo estimaatit eivät ole täsmällisiä, vaan niihin liittyy luotettavuusrajat. Mitä suurempi simuloinnin otosmäärä on, sitä tarkempi tulos saadaan. Jos oletetaan että malli on oikea, tilastolliset virheet on saatavissa niin pieniksi kuin vain halutaan suorittamalla riittävän suuri määrä otoksia. Valitettavasti simuloinnin kustannukset myös kasvavat otosmäärän suhteessa ja kustannukset tulevat liian suuriksi, jos malli on monimutkainen ja jos on tutkittava syöttösuureiden useita erikombinaatioiden vaikutuksia kuten tässä risteys-simuloinnissa. Täten tavoitteena on suorittaa vain minimimäärä simulointiajoja halutun tarkkuusasteen saavuttamiseksi. Monte Carlo-menetelmälle on yleinen piirre, että tuloksen todennäköinen virhe riippuu otoksen suuruudesta. Tuloksen todennäköisyyden virheen pienentäminen otosta suurentamalla johtaa helposti kohtuuttomiin laskenta-aikoihin. Tämän vuoksi on kehitetty lukuisia varianssia pienentäviä tekniikoita, kun Monte Carlo-menetelmän sovellutusten määrä on kasvanut valtavasti tietokoneiden suorittamiskyvyn myötä /12, 13, 14, 19, 28/.

On etsittävä kustannusten ja ajan suhteen minimiin johtava menetelmä, jolla kuitenkin voidaan erottaa eri vaihtoehtoisten toimintojen vaikutus toisistaan jollakin varmuudella. Täsmällisemmin ilmaistuna tämä tarkoittaa, että Monte Carlo-simuloinnin tavoite on tutkittavan suureen Y estimointi tietyllä luotettavuusvälillä minimikustannuksiin. Jos Y on normaalijakautunut, luotettavuusväli on ilmaistavissa Moyn /28/ mukaan

$$P\{\bar{Y} - t_{\alpha/2} s/\sqrt{n} < E(Y) < \bar{Y} + t_{1-\alpha/2} s/\sqrt{n}\} = 1 - \alpha, \quad (5-1)$$

jossa \bar{Y} ja s vastaavat otoksen keskiarvoa ja standardipoikkeamaa ja t_{α} on t -jakautuman α -osuus, kun vapausaste on $n - 1$.

Oletetaan että normaalia otostekniikkaa käyttäen saavutetaan tietty luotettavuusväli kustannuksin C_1 ja toista otostekniikkaa käyttäen yhtä luotettavien tulosten saamisen kustannukset ovat C_2 . Kun verrataan näitä tekniikoja, toista tekniikkaa käyttäen saatava hyöty on $(C_1 - C_2) / C_1$.

Jos oletetaan että $n:n$ havainnon otoksen kustannus riippuu n :stä molemmissa tapauksissa: esim. $C_1 = k_1 n_1$ ja $C_2 = k_2 n_2$, jossa n_1 ja n_2 vastaavat otosmääriä. Tällöin kaavasta 5-1 saadaan, kun molempia tekniikoita käytetään samalla varmuusvälillä:

$$\frac{n_2}{n_1} = \left(\frac{t_{\alpha/2,2} s_2}{t_{\alpha/2,1} s_1} \right)^2, \quad (5-2)$$

jossa on samat symbolit kuin kaavassa 5-1 ja indeksit viittaavat vertailtavana oleviin tekniikoihin. Tekniikkaa 2 käyttäen saatava hyöty on nyt ilmaistavissa:

$$\text{hyöty (\%)} = 100 \left\{ 1 - \frac{k_2}{k_1} \left(\frac{t_2 s_2}{t_1 s_1} \right)^2 \right\} \quad (5-3)$$

Lausekkeen 5-3 mukaan, jos otoskustannukset ovat molemmille tekniikoille samat ja jos $t_1 = t_2$, saatava hyöty on sama kuin varianssin pienentymisen määrä.

Monte Carlo-sovellutuksia varten erikoisesti keksittyjä otostekniikkoja ovat: importance sampling, antithetic-variate sampling, sekutive sampling, systematic sampling, Roussian roulette, joista kahta ensinmainittua mm. Kalli on käsitellyt artikkelissaan /19/. Hammerslayn /14/ mukaan joillakin erikoisilla otostekniikoilla on raportoitu saavutetun efektiivisen varianssin pienenemistä jopa $10^4 \dots 10^6$ kertaluokkaa olevilla tekijöillä. Tavallisimpien systeemien simuloinneissa on varianssin pienennys-tekniikoita käyttäen Moyn /28/ mukaan saavutettavissa kolmasosan säästö.

5.3

Työn nykyvaihe ja jatkotutkimuksista

Tämän työn tavoitteeksi asetettu risteysliikenteen simuloimallin konstruointi ja sen ohjelmointi toimivaksi tietokoneohjelmaksi on suoritettu. Lisäksi simulaattorin toimintalogiikalle ja osamalleille on suoritettu validiteettitestauksia niin monia, kuin niitä on ollut mahdollista toteuttaa saatavissa olevaan niukkaan testiaineistoon nähden. Mallin eri parametrien vaikutusta eri liikennetoimintoihin on myös tutkittu. Esimerkkiajot on suoritettu kuvitellussa risteyksessä, jonka ominaisuuksista osa on todellista. Täten tälle työlle asetetut tavoitteet on saavutettu.

Jatkotutkimuksena on seuraavaksi suoritettava simulaattorin kokonaisvalidisointi kohdassa 5.1 esitettyjen kriteerien mukaan, mikä edellyttää sitä ennen suoritettavaksi eräiden liikennekäyttäytymistietojen keräämisen. Lisäksi mallissa on käytetty joitakin "hihasta vedettyjä" todennäköisyysjakautumia monien kirjallisuudesta löydettyjen jakaantumien lisäksi, joiden oikeellisuus ja soveltuvuus Suomen oloihin olisi liikennetekniikan asiantuntijoiden tarkistettava ja jotka mahdollisesti vaativat kenttätutkimuksia. Tämän työn atk-ratkaisun suunnittelu- ja ohjelmointitoimeksiantoonhan liikennetekniikkaan liittyvien jakautumien etsiminen ja tutkiminen eivät kuuluneet. Keskeisimmät jakautumat, jotka vaatisivat lisätutkimuksia ovat:

1. konfliktien määrä- ja vakavuusastejakautumat
2. liikennevirtaa koskevat ominaisuudet (nopeudet, aikavälit jne.).
3. jalankulkijoiden esiintymistodennäköisyysjakautumat
4. risteyksen välityskykyä koskevat suureet (jononpituudet, läpikulkuajat)
5. havaitsemista ja päätöksentekoa koskevat jakautumat
6. näköesteiden ja mahdollisten muiden suureiden merkitse-

vyys konfliktitilanteihin esimerkiksi onnettomuustutkimusten perusteella.

Mallin kokonaisvalidisoinnin jälkeen malli on valmis käytettäväksi muutaman katuristeyksen konfliktitutkimukseen, jossa kannattaa kiinnittää huomio testitulosten analysointiin, simulointihypoteesien testaukseen, sekä varianssi-probleemaan ja otoksen määrään kohdassa 5.2 esitettyjen perusteiden vuoksi.

Työssä on kehitetty yleinen liikenneristeyssimulaattori, jota kehitettäessä on pidetty mielessä ensi sijaisesti sen soveltamista konflikti- ja onnettomuustutkimuksiin. Simulaattori on suunniteltu liikennevaloin ohjatun mielivaltaisen muotoisen katuristeyksen liikenteen simulointiin, mutta yhtä hyvin se soveltuu käytettäväksi, vaikka liikennevalo-ohjaus puuttuu tai risteys on eritasoristeys. Liikenne voi koostua kuudesta eri komponentista jalankulkijat mukaanluettuna:

Simulaattorin toteutus on suoritettu kahdessa vaiheessa. Ensiksi systeemin (, joka koostuu ajoradoista, ajoneuvoista, kuljettajista ja liikenne- ja ympäristötekijöistä) mikroaspektit on kuvattu liikennevirtateorian ja sovelletun liikennepsykologian mukaisilla matemaattisilla malleilla.

Toiseksi suoritettiin näiden matemaattisten mallien ja niitten soveltamista ohjaavan käyttäytymislogiikan koodaus tietokone-ohjelmaksi, joka prosessoi osamallit saaden aikaan abstraktin systeemin dynaamisen ratkaisun.

Ohjelmalogiikka on jaettavissa kahteen osaan: ylempänä hierarkisella tasolla olevat käyttäytymismallit ohjaavat etenemismallien soveltamista. Ohjelma on tehty modulaariseksi, joten sitä on helppo haluttaessa muuttaa. Ohjelmointikielenä on käytetty FORTRAN:ia ja tietokoneena UNIVAC 1108:aa, mutta ohjelmaa koodattaessa on otettu huomioon sen sovittaminen muille koneille, esim. pientietokoneille.

Simulaattorin tulostus keskittyy konfliktitilanteista rekisteröityjen tietojen ja risteysvälityskykyä mittavien jononpituuksien tulostukseen. Tietyistä konfliktitilanteista on mahdollista saada erillinen ilmavalokuva muistuttava tulostus rumpupiirturilaitetta hyväksikäyttäen. Simulaattorista on kuitenkin helposti saatavissa lisätulostusta mitä moninaisimmista liikennesuureista.

Simulointimallin toteuttaminen on osoittanut, että Monte Carlo - simulointitekniikka soveltuu hyvin käytettäväksi kompleksisten liikenneilmiöiden kuten konfliktitilanteiden analysoinnissa. Mallin toimintakyky riippuu tietenkin systeemin funktiokomponentteja kuvaavien osamallien ja systeemin saamien syöttötietojen tarkkuuksista. Valitettavasti mallin kokonaisvalidisointia ja kalibrointia todelliseen risteykseen ei voitu suorittaa kenttätutkimuksista saatavan informaation puuttumisen vuoksi.

Liikennesimulaattorin tehokkuuden mittana usein käytetty suhde: tietokoneaika/reaaliaika ei yksistään ilmaise paljoakaan. Se riippuu voimakkaasti simuloitavan risteyksen koosta ja kompleksisuudesta sekä ajoneuvotiheydestä. Tässä työssä kehitetty simulaattori näyttää kuluttavan tietokoneaikaa enintään 1/10 reaaliajasta, kun testiajoissa simuloitiin ruuhkaliikennettä suuressa, 13-sisääntulokaistaisessa risteyksessä.

7. KIRJALLISUUSVIITTEET JA LISTAUKSET

7.1 Lähdekirjallisuus

7.1.1 Tekstissä referoitu lähdekirjallisuus

1. Ahrens J.H., Dieter U., Grube A., Pseudorandom numbers. A new proposal for the choice of multipliers. Computing 6 (1970) 1-2, s. 121...138.
2. Andersin Hans E., Sulonen Reijo, Simulerinteknik. Lund, Studentlitteratur, 1972.
3. Ashworth Robert, Delays to pedestrians crossing a road. Traffic Engineering & Control 13 (1971) 3, s. 114...115.
4. Beilby M.H., Road traffic simulation on a small computer. The Computer Journal 15 (1972) 2, s. 134...137.
5. Blake K. and Gordon G., Systems simulation with digital computers. IBM Systems Journal 3 (1964) 1, s. 14...20.
6. Blum A.M., A general purpose digital traffic simulator. Simulation 14 (1970) 1, s. 9...25.
7. Brill Edward A., A car-following model relating reaction times and temporal headways to accidents frequency. Transportation Science 6 (1972) 4, s. 343...353.
8. Carlson William L., Application of simulation to assessment of component changes on rear and highway accidents. Transportation Science 6 (1972) 1, s. 1...14.
9. Farbes T.W., Human factor considerations in traffic flow theory. Washington 1963. Highway Research Board Record No. 15, s. 60...66.
10. Fox P., Lehman F.G., A digital simulation of car-following and overtaking. Washington 1967. Highway Research Board. Highway Research Record No. 199, s. 33...41.
11. Gerlough D.L., Wagner F.A., Improved criteria for traffic signals at individual intersections. Washington 1967. Highway Research Board. National Co-operative Highway Research Program Report No. 32.

12. Hahn G.J., Sample sizes for Monte Carlo simulation. IEEE transactions on Systems, Man and Cybernetics. SMC-2 (1972) 5, s. 678...680.
13. Halton John H., A retrospective and prospective survey of the Monte Carlo method. Siam Review 12 (1970) 1, s. 1...63.
14. Hammerslay J.M., Handscomb D.C., Monte Carlo Methods. London, Methnen & Co, 1967.
15. Herman R., Montroll E.W., Potts R.B., Rothery R.W., Traffic dynamics. Analysis of stability in car following. Operations Research 7 (1959) 1, s. 86...106.
16. Häkkinen Sauli., Liikenteen ohjauksen psykologiaa. Helsinki 1965. Talja. Julkaisematon.
17. Häkkinen Sauli, Estimation of distance and velocity in traffic situations. Helsinki 1963. Työterveyslaitos. Reports from the Institute of Occupational Health No. 3.
18. Kallberg Harri, Liikenteen simulointi. Helsinki 1971. Teknillinen korkeakoulu. Lisensiaattityö.
19. Kalli Heikki, Monte Carlo-menetelmän sovellutuksia. Tutkimus ja Tekniikka (1974) 3, eripainos.
20. Kidd Edwin A., Laughery K.R., Computer model of driving behaviour. The highway intersection situation. Washington 1966. Highway Research Board. Highway Research Record No. 118, s. 96...97.
21. Klijnhout J.J., A digital simulation model for signal intersections with traffic lights. Traffic Engineering & Control 13 (1971) 4, s. 147...150,153.
22. Kottler C.F., McGill R., Using minicomputers to reduce problem solving costs. Instruments and Control Systems, April 1973, eripainos.
23. Larsson Uno, Lundin Rolf, Urban traffic simulation. Göteborg, BAS, 1971.

24. Laski J.G., On time structure in (Monte Carlo) simulations. Operational Research Quarterly 16 (1965) 3, s. 329...339.
25. Lave Roy E.Jr., Timekeeping for simulation. The Journal of Industrial Engineering 18 (1967) 7, s. 389...394.
26. May A.D. and Keller H.E.M., Non-integer car-following models. Washington 1967. Highway Research Board. Highway Research Record No. 199, s. 19...32.
27. Mihran Arthur G. Simulation. Statistical Foundations and Methology. London, Academic Press, 1972.
28. Moy W., Sampling techniques for increasing the efficiency of simulations of queueing systems. Evanston Illinois 1965. Northwestern University. Dissertation.
29. Naylor T.H., Balintfly J.L., Burd ick D.S., Chu K., Computer simulation techniques. London, John Wiley, 1966.
30. Newell G.F., Application of queueing theory. London, Chapman & Hall, 1971
31. Nyyssönen Reino, Psykologinen etuajo-oikeus ja risteyksen vaarallisuusaste. Helsinki 1973. Liikenneturva, tutkimusosasto. Julkaisematon.
32. Olson P.L., Rothery R.W., Driver response to the amber phase of traffic signals. Traffic Engineering & Control 32 (1962) 5, s. 17...21.
33. Russam K., Sabey B.E., Accidents and traffic conflicts at junctions. London 1973. Transportation and Road Research Laboratory. TRRL Report LR 514.
34. Seddon Philip A., An other look at platoon dispersion. 1. Kinematic wave theory, 2. Differsion theory, 3. The recuerence relationship. Traffic Engineering & Control 13 (1972) 8, s. 332...336. 9, s. 388...390. 10,s. 442...444.
35. Seddon Philip A., A program for simulatind the dispersion of platoon of road traffic. Simulation 18 (1972) 3, s. 81...90.

36. Sinha Kumares Chandra, The development of a digital simulator for the analysis of freeway traffic phenomena. Connecticut 1968. University of Connecticut. Dissertation.
37. Solberg P, Oppenlander J.C., Lag and gap acceptances at stop-controlled intersections. Washington 1966. Highway Research Board. Highway Research Record No. 118, s. 48...67.
38. Tolle J.E., Probabilistic aspects of safety in traffic flow. Traffic Engineering & Control 14 (1973) 10, s. 489...490.
39. Wagner F.A. Jr., An evaluation of fundamental driver decisions and reaction at an intersection. Washington 1966. Highway Research Board. Highway Research Record No. 118, s. 68...84.
40. Virjo Antti, Eräiden simulointikielten vertailu. Helsinki 1972. Valtion tietokonekeskus. VTKK raportti 1972-05-10.

7.1.2 Referoimaton lähdekirjallisuus

1. Ahola Erkki, Liikenteen simulointi. Helsinki 1971. Teknillinen korkeakoulu. Matematiikan erikoistyö.
2. Allsop Richard E., Sigset - A computer program for calculating traffic signal settings. Traffic Engineering & Control 13 (1972) 2, s. 58...60.
3. Beilby M.H., Lewis P.A., Computer simulation of road traffic. Traffic Engineering & Control 13 (1971) 7, s. 311.
4. Black B.C., Gazis D.C., Real-time traffic flow optimization. IBM Systems Journal 10 (1971) 3, s. 217...231.
5. Brill Edward A., A model of a traffic jam behind a bottleneck. Operations Research 20 (1972) 4, s. 791...799.
6. Davies N.R., Decision tables in discrete-system simulation. Simulation 19 (1974) 2, s. 39...44.
7. Dunne Michael C., Potts Renfrey B., Algorithm for traffic control. Operation Research 12 (1964) s. 870...881.
8. Dunne Michael C., Traffic delay at a signaled intersection with binomial arrivals. Transportation Science 1 (1967) 1, s. 24...31.
9. Fedotkin M.A., Traffic control at an intersection with periodic random flux of arriving vehicles. Engineering Cybernetics 3 (1969) 3, s. 57...65.
10. Gazis Denos C., Optimum control of a system of oversaturated intersections. Operations Research 12 (1964) , s. 815...831.
11. Gazis Denos, Herman Robert, Maradudin Alexei, The problem of the amber signal light in traffic flow. Traffic Engineering 30 (1960) 10, s. 19...26, 53.

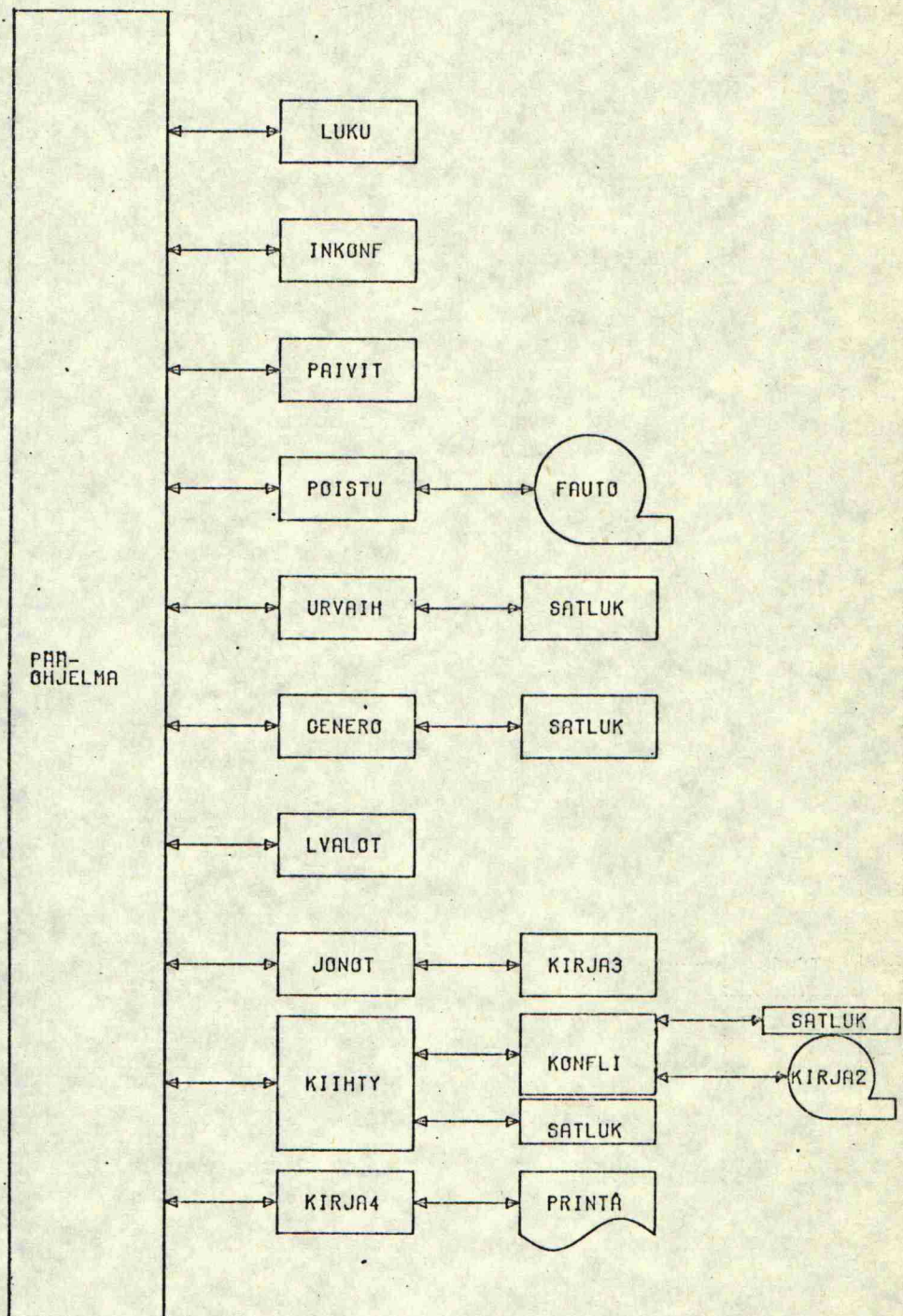
12. Green D.H., Hartley M.G., The simulation of some simple control policies for a signalized intersection. Operational Research Quarterly 17 (1966) 3, s. 263...278.
13. Haight F.A., Mathematical Theories of Traffic Flow. New York, Academic Press, 1963.
14. Häkkinen Sauli, Large scale field experiments on driver behaviour. Zeitschrift für verkehrssicherheit 20 (1974) 2, s. 75...93.
15. Jones Robert A., Computer simulation in conflicting vehicle storage problem. Traffic Engineering 39 (1969) 5, s. 42...44.
16. Kiviat Philip, Digital computer simulation. Computer programming languages. Washington 1970. Rand Corporation. Rand Corporation Memorandum RM-5378-PR.
17. Kleinecke D.C., Discrete time queues at a periodic traffic light. Operations Research 12 (1964) s. 809...814.
18. Lehoczy J.P., Traffic intersection control and zero-switch queues under conditions of Markov chain dependence input. Journal of Applied Probability 9 (1972) 2, s. 382...395.
19. Martin-Löf Anders, Computation of an optimal control for a signalized traffic intersection. Transportation Science 1 (1967) 1, s. 1...5.
20. Morris R.W.J., Pack-poy P.G., Intersection control by vehicle actuated signals. Traffic Engineering & Control 9 (1967) 6, s. 288...293.
21. Newell Gordon F., Approximation methods for queues with application to the fixed cycle traffic light. SIAM Review 7 (1965) 2, s. 223...240.
22. Newell Gordon F., Properties of vehicle actuated signals: I One-way streets. Transportation Science 3 (1969) s. 30...52.
23. Potts Renfrey B., Traffic delay at a signalized intersection with binomial arrivals. Transportation Science 1 (1967) , s. 126...128.

24. Powner E.T., Hartley M.G., Heath F.G., Green D.H., Road traffic simulation employing a hardware approach philosophy and design considerations. *Simulation* 15 (1970) 3, s. 113...118.
25. Rumar Kåre, Berrrund Ulf, Overtaking performance under controlled conditions. Uppsala 1973. University of Uppsala. Department of psychology report 148.
26. Rumsey A.F., Hartley M.G., Simulation of a pair of intersections. *Traffic Engineering & Control* 13 (1972) 11-12, s. 522...525.
27. Sagen Ragnvald, Traffic simulation with cathode ray tube output. Trondheim 1967. The Technical University of Norway. Department of Road Construction and Traffic Engineering. Report no. 8.
28. Sagi George S., Campbell Lowell R., Vehicle delay at signalized intersections, Theory & practise, *Traffic Engineering* 39 (1969) 5, s. 32...40.
29. Solin M., Kuljettajan valppaus, väistämiskonfliktit ja onnettomuudet psykologisesti etuajo-oikeutetuissa risteyksissä. Helsinki 1969. Julkaisematon.
30. Story C.E.R., Simulation of traffic by digital computer. *Traffic Engineering & Control* 11 (1970) 10, s. 464...467.
31. Webber D., Some applications of computers in traffic engineering. *Traffic Engineering & Control* 13 (1972) 10, s. 445...447.
32. Wigan M.R., Bamford T.J., A computer network simulation of different methods of traffic restraint. Crowthorne 1973. Transportation and Road Research Laboratory. TRRL Report RL 566.
33. Wigan M. Ramsay, Applications of simulation to traffic problems. *Simulation* 14 (1970) 3, s. 135...136.

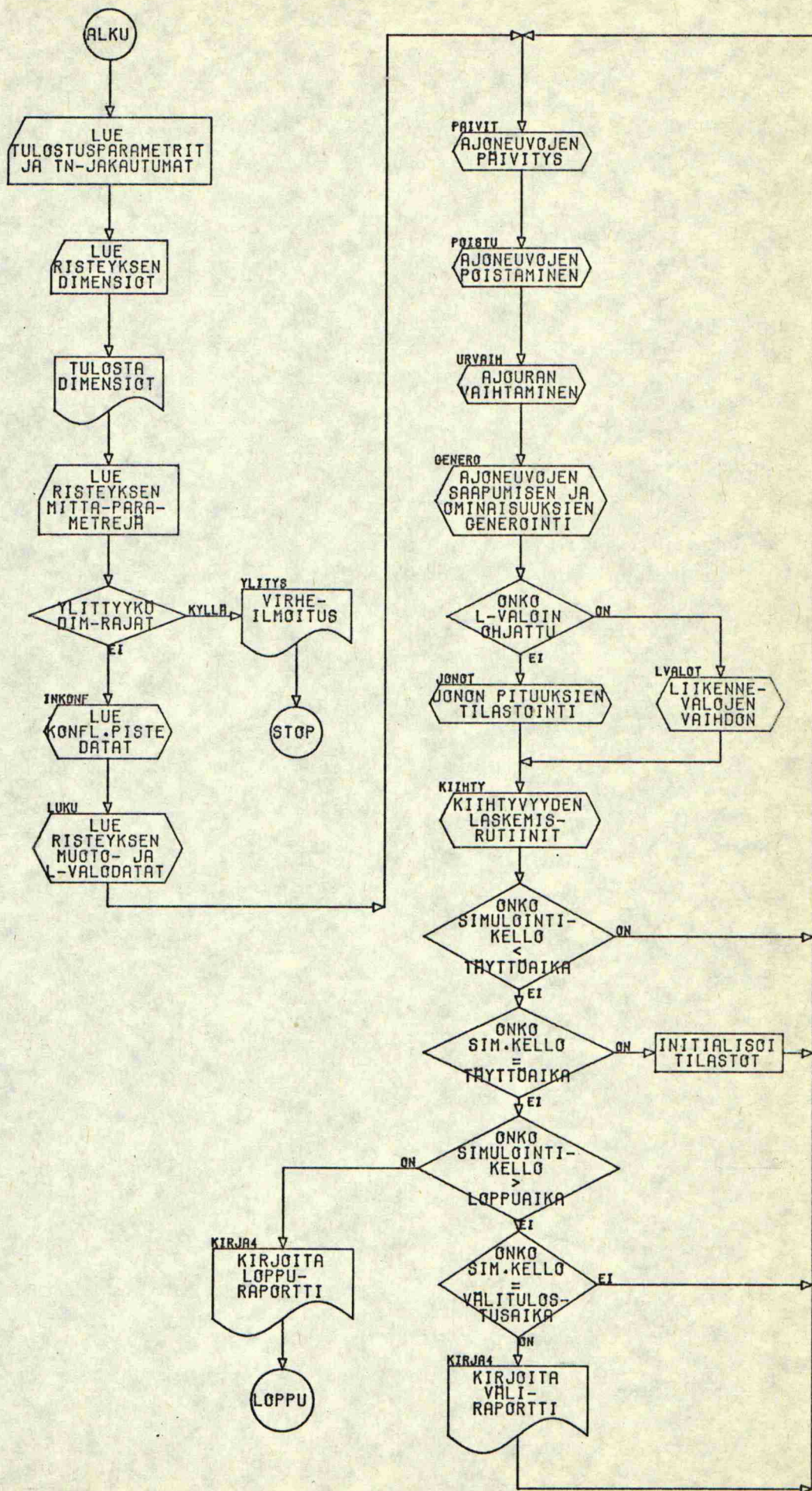
7.2

Ohjelmayksiköiden kulkukaaviot

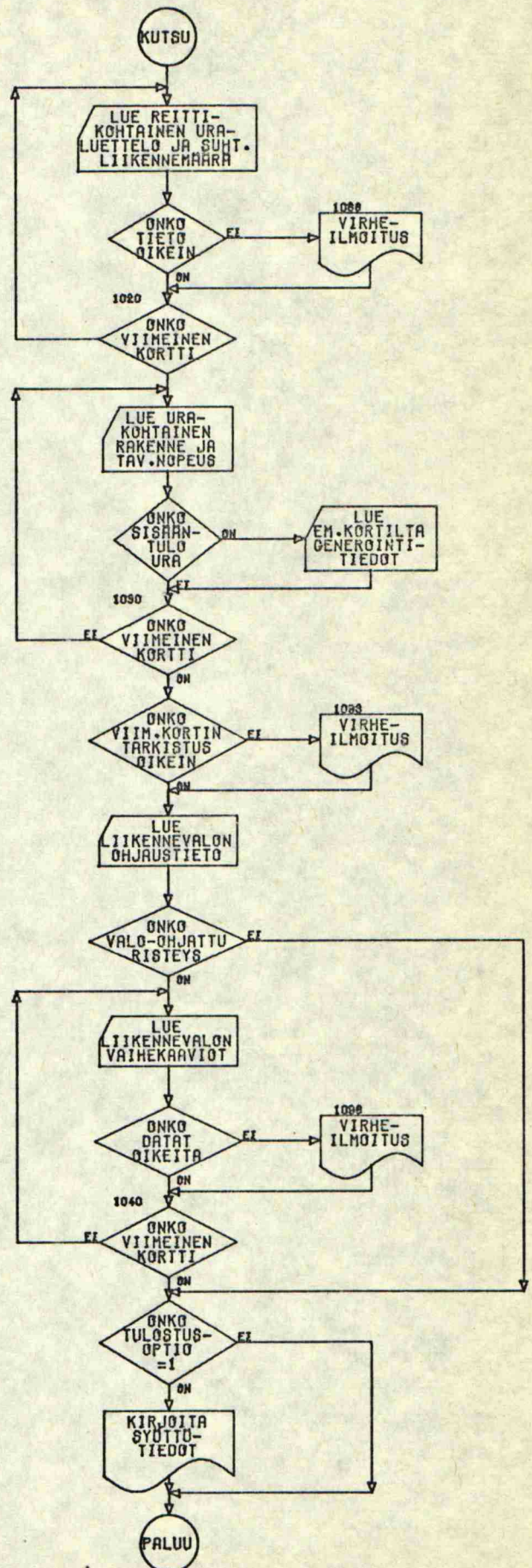
KUVA 4-3. OHJELMISTON KOKONAISLOGIIKKA



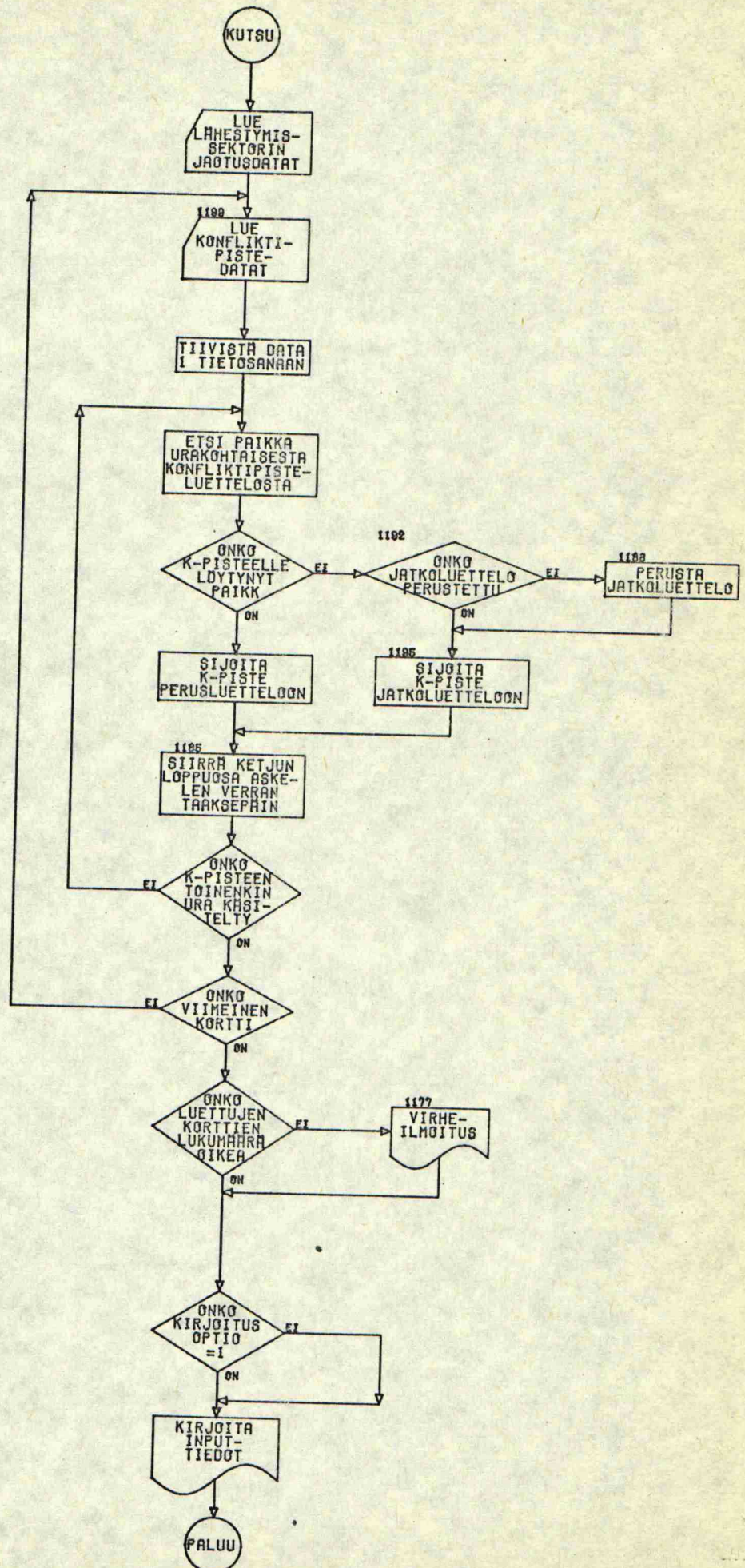
KUVA 4-4. PÄÄOHJELMAN KÄSITTELYLOGIIKKA



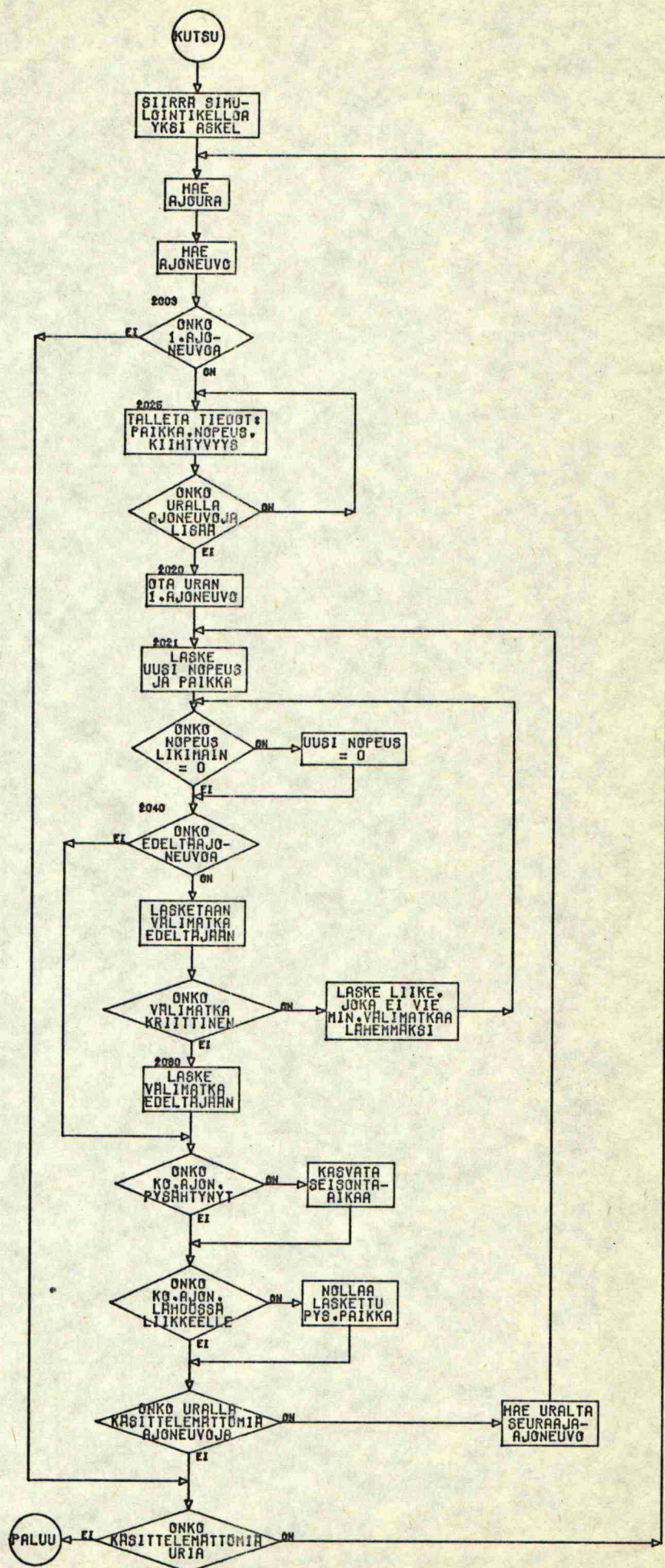
KUVA 4-5. ALIOHJELMAN LUKU KULKUKAAVIO



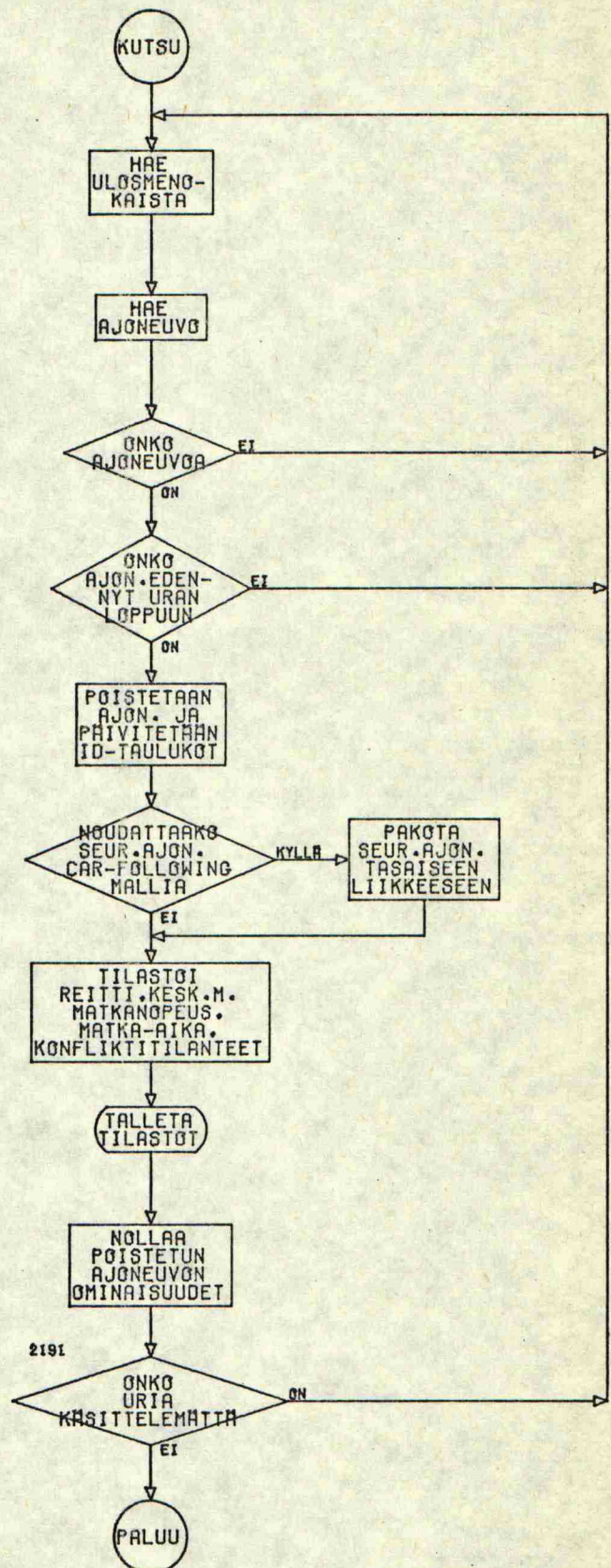
KUVA 4-6. ALIOHJELMAN INKONF KULKUKAAVIO



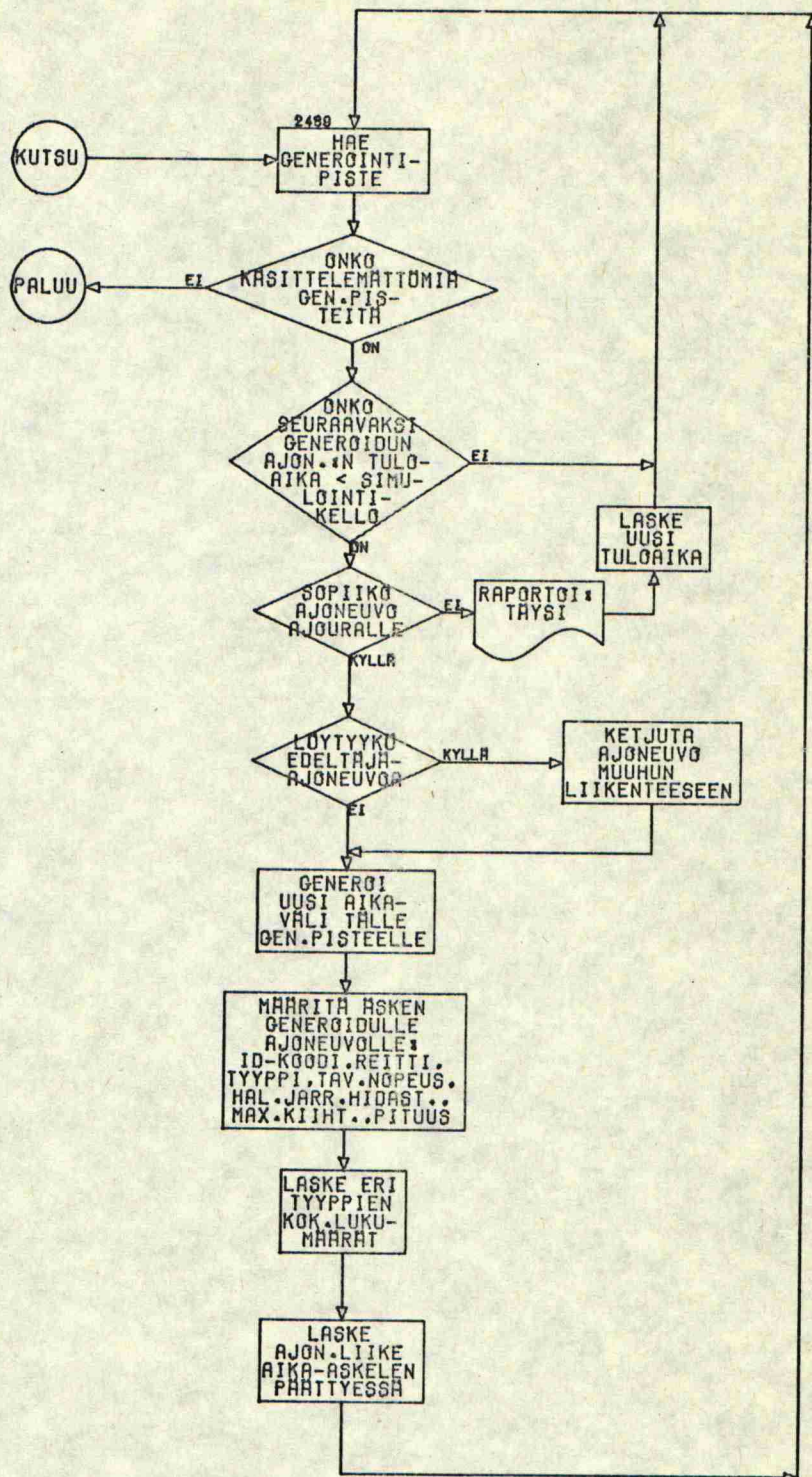
KUVA 4-7. ALIOHJELMAN PAIVIT KÄSITTELYLOOGIIKKA



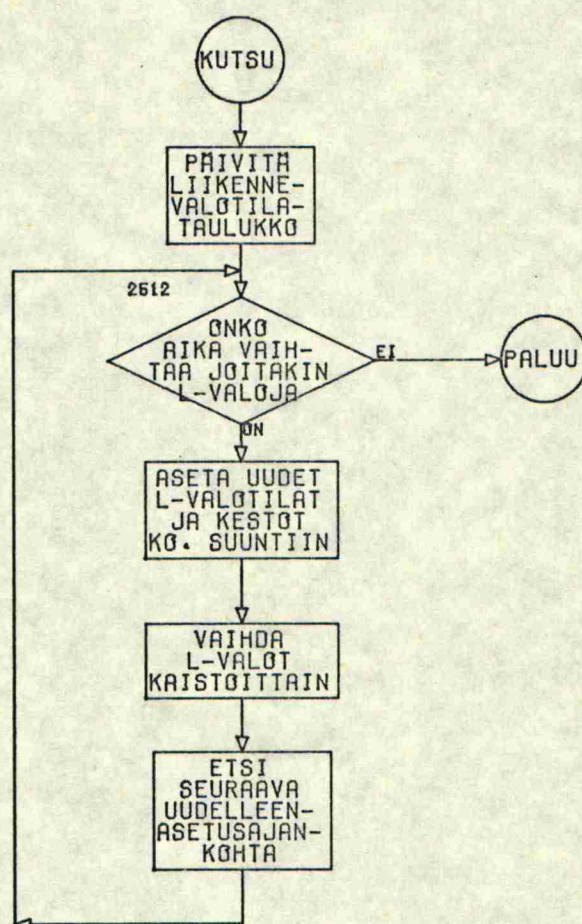
KUVA 4-8. ALIOHJELMAN POISTU KÄSITTELYLOGIIKKA



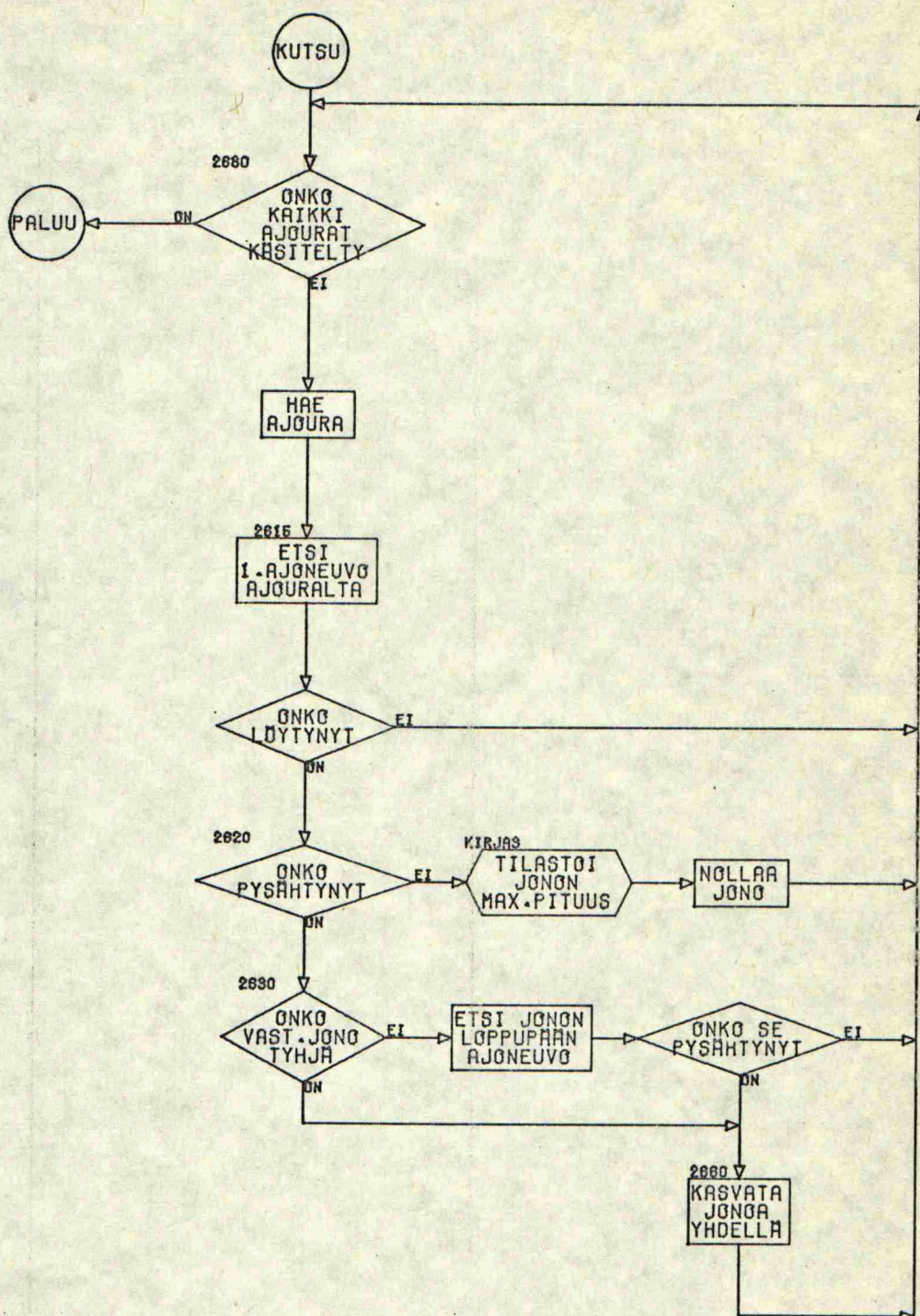
KUVA 4-10. ALIOHJELMAN GENERO KÄSITTELYLOGIIKKA



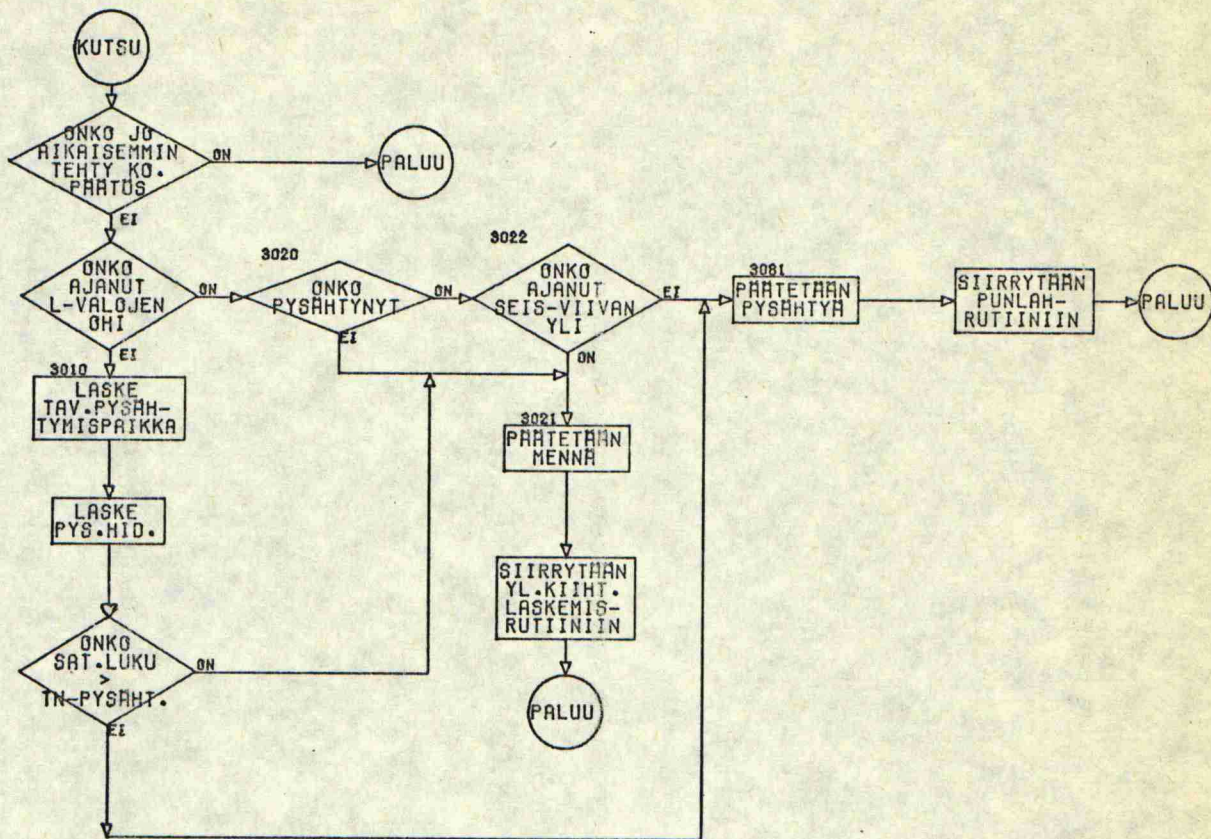
KUVA 4-11. ALIOHJELMAN LVALOT KÄSITTELYLOGIIKKA



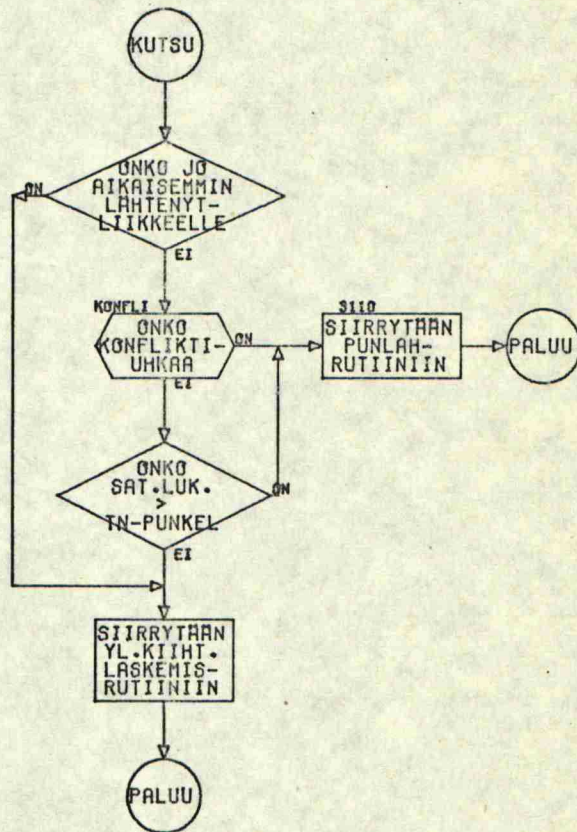
KUVA 4-12. ALIOHJELMAN JONOT KÄSITTELYLOGIIKKA



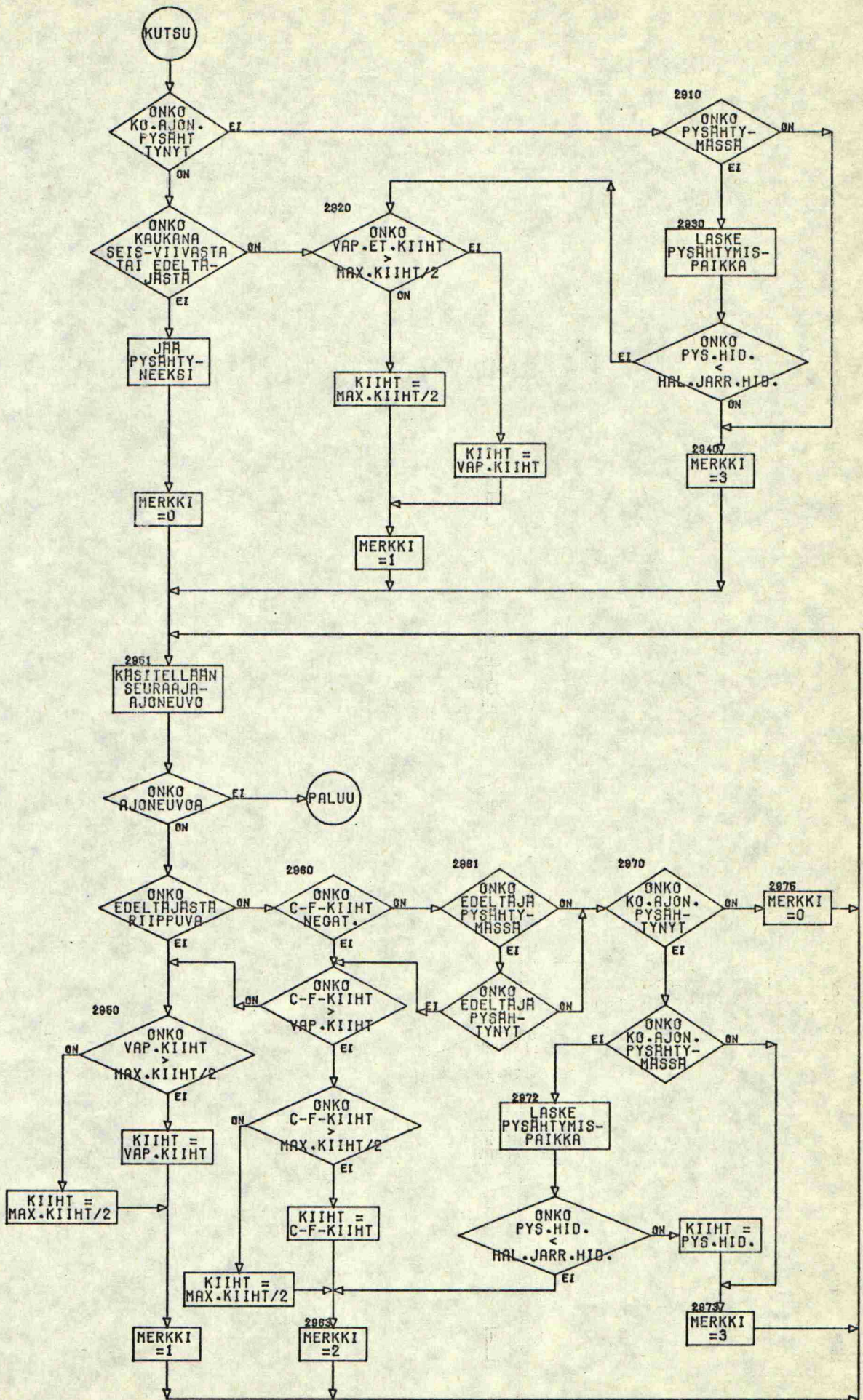
KUVA 4-14. ALIRUTIININ VIHKEKÄSITTELYLOGIIKKA



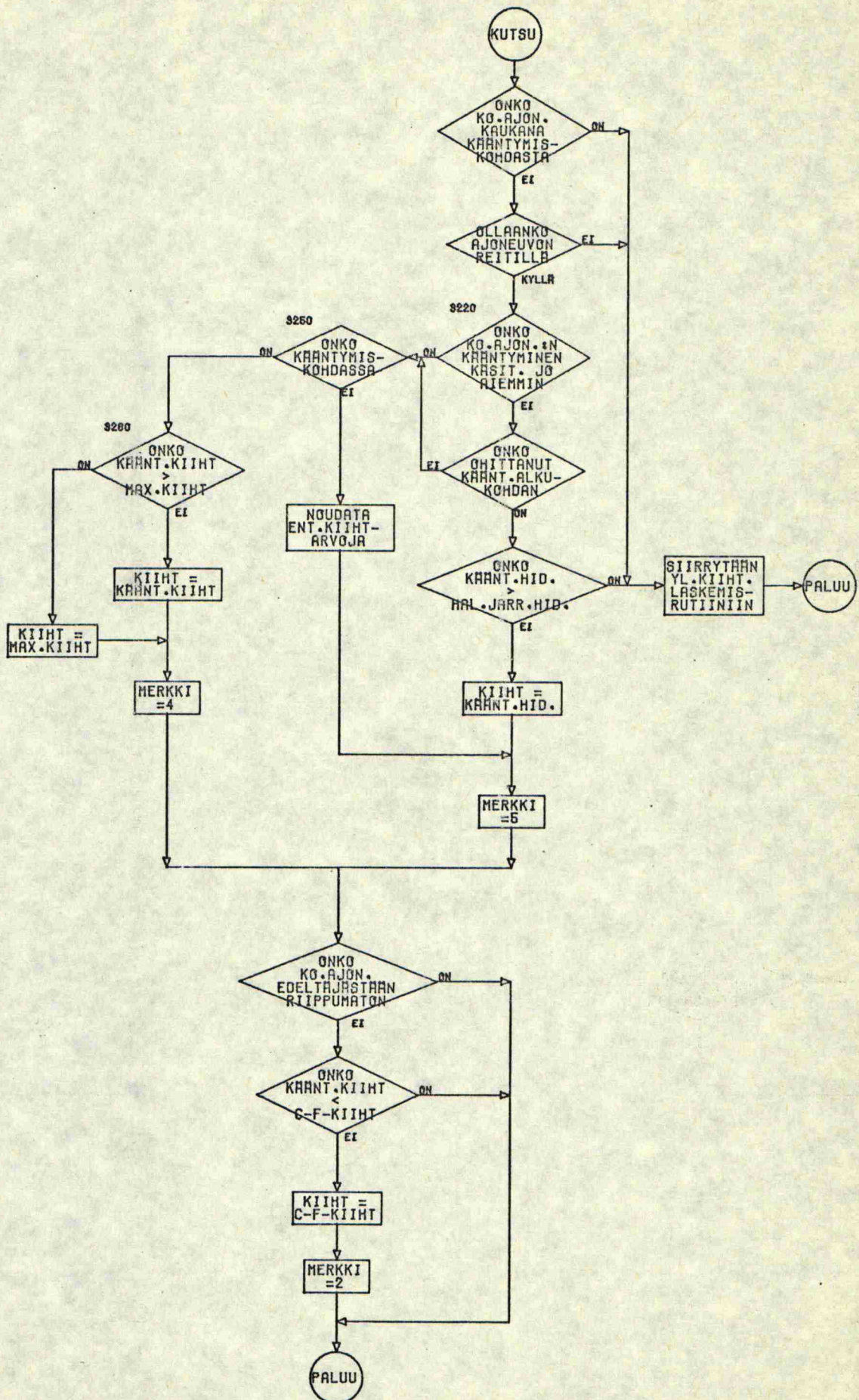
KUVA 4-15. ALIRUTIININ PUNKEL KÄSITTELYLOGIIKKA



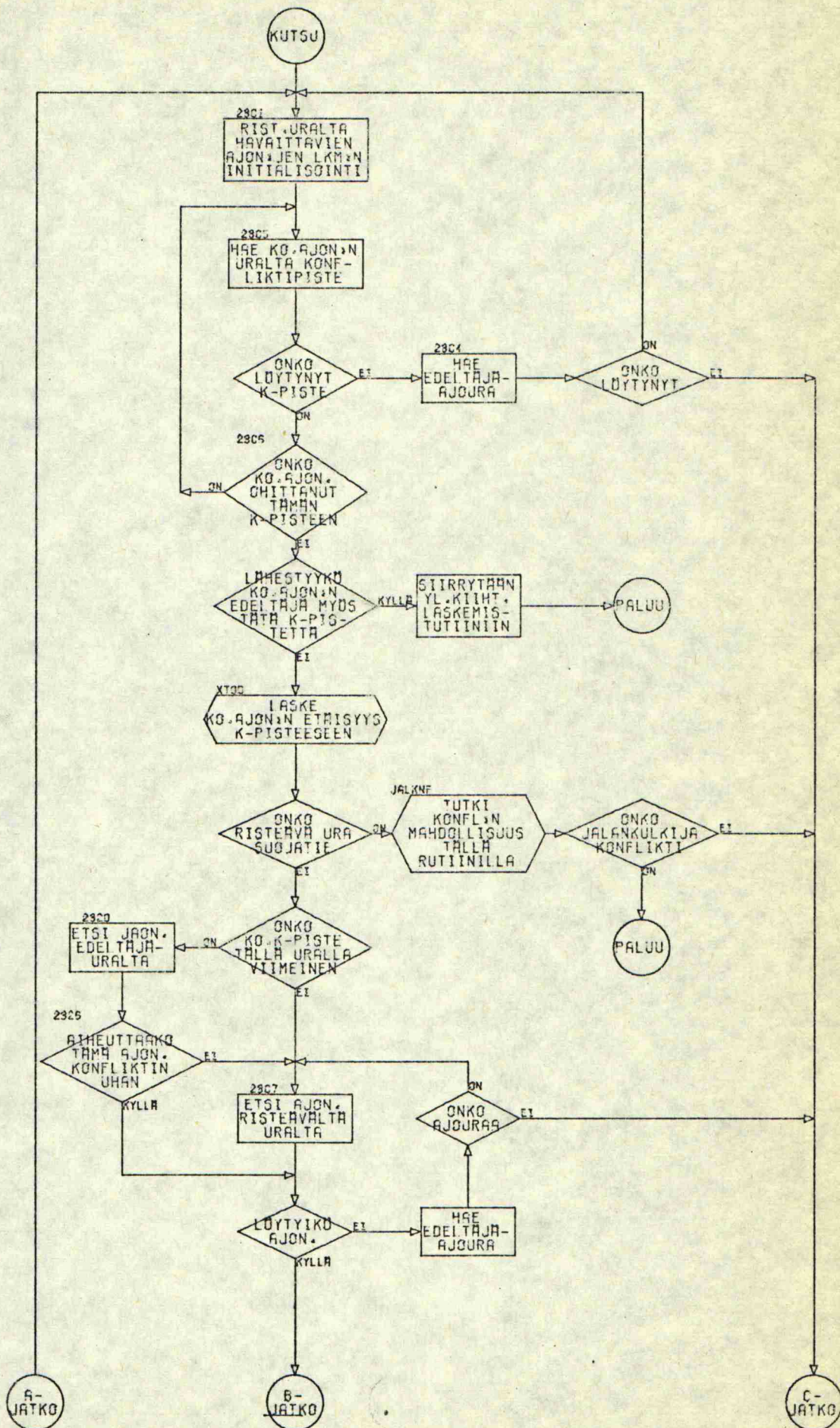
KUVA 4-16. ALIRUTIININ PUNLAH KÄSITTELYLOGIIKKA



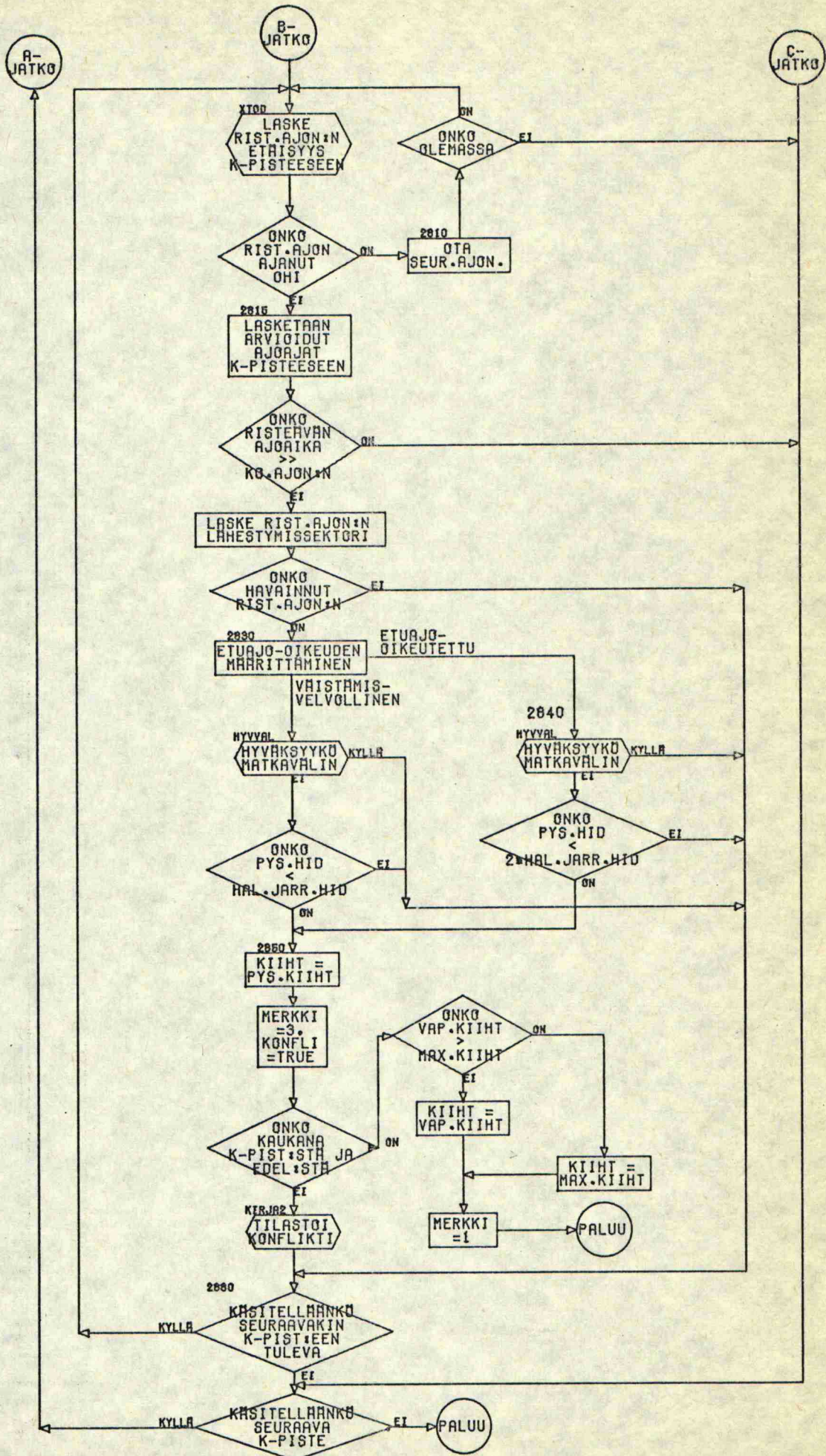
KUVA 4-17. ALIRUTIININ KAANTY KÄSITTELYLOGIIKKA



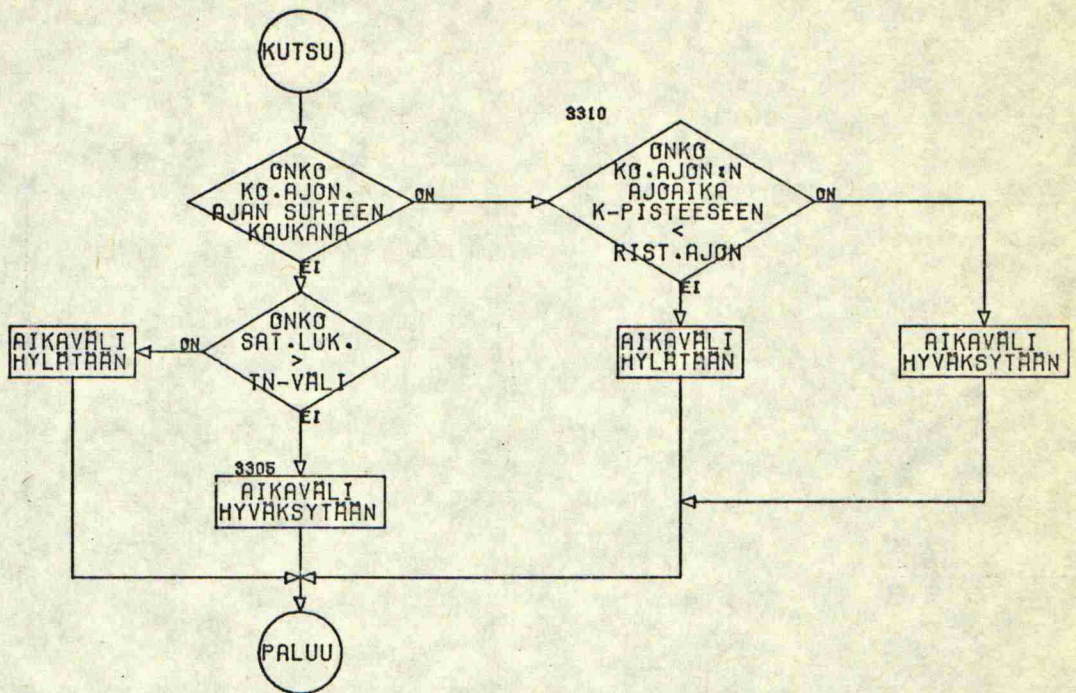
KUVA 4-18. ALIOHJELMAN KONFLI KASITTELYLOGIIKKA (OSA 1)



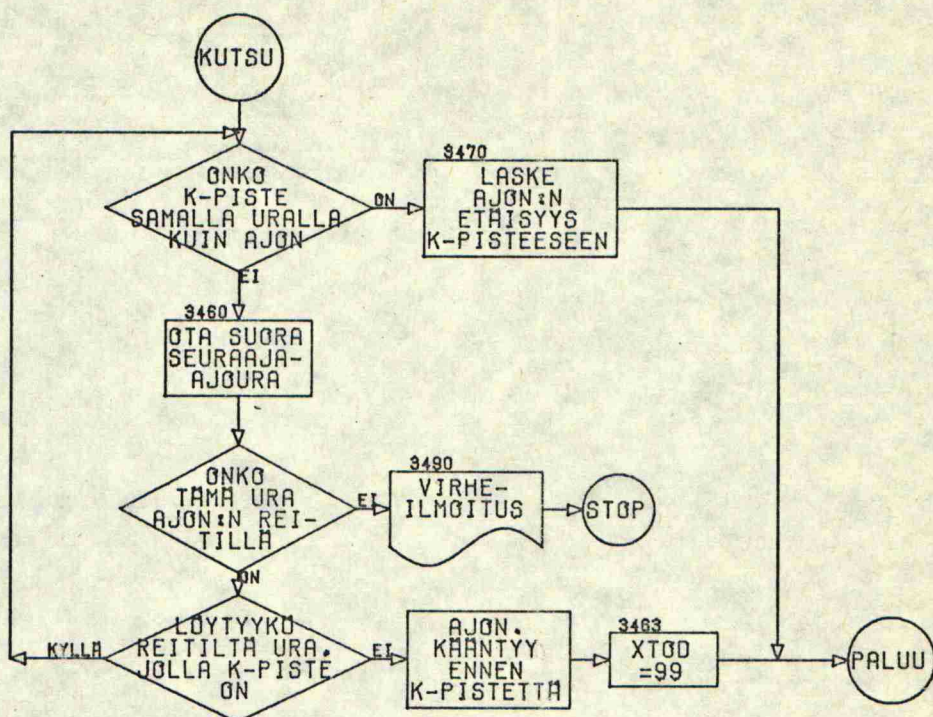
KUVA 4-18. ALIOHJELMAN KONFLI KÄSITTELYLOGIIKKA (OSA 2)



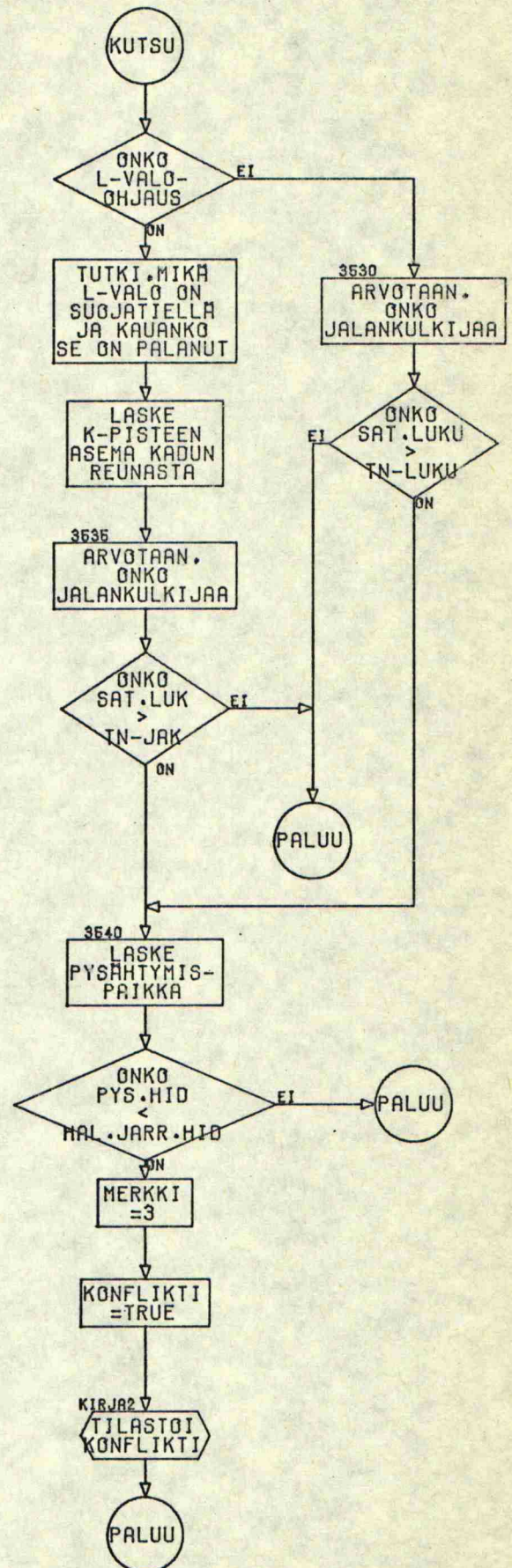
KUVA 4-19. ALIFUNKTION HYVVAL KÄSITTELYLOGIIKKA



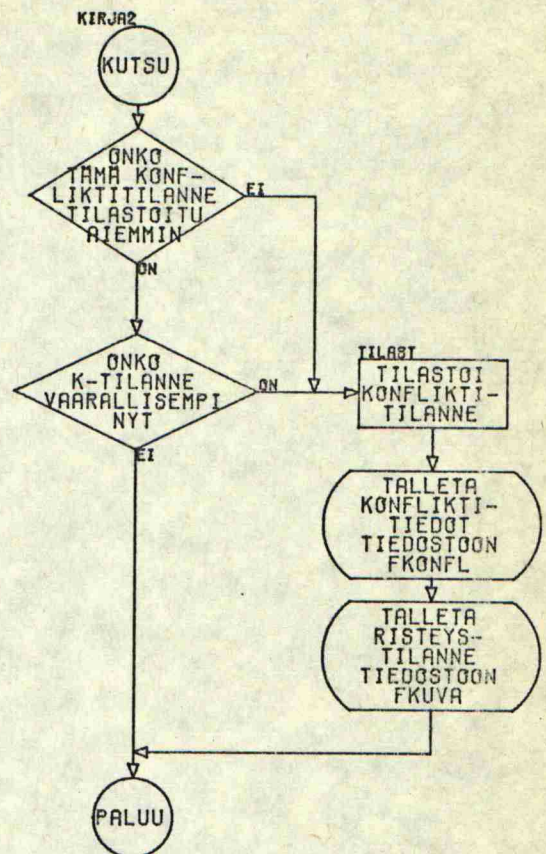
KUVA 4-20. ALIFUNKTION XTOD KÄSITTELYLOGIIKKA



KUVA 4-21. ALIRUTIININ JALKNF KÄSITTELYLOGIIKKA



KUVA 4-22. ALIOHJELMAN KIRJA KÄSITTELYLOGIIKKA




```

1*      PARAMETER NOM=5,NAU=100,NUR=42,NSK=13,NUK=12,NRE=25,ITV=3,NKON=72,      000000
2*      1      NKON2=2*NUR,LVT=4,IVAL=NSK+8,NAUT3=3*NAU      000000
3*      PARAMETER DPAI=1,DURV=1,DGEN=1,DKII=1,LPAI=1,LURV=1,LGEN=1,LKIT=1      000001
4*      REAL X(NAU),V(NAU),A(NAU,ITV),ALFA(NAU),TPYS(NAU),EPIT(12,NAU),      000001
5*      1      XPYS(NAU),AU(NAU,NOM),XURA(NUR),LAMBDA(5,NSK),VURA(NUR),      000001
6*      2      TGEN(NSK),PROS(NRE),XVALI(NAU),TNKFLT(20),TNVALI(20)      000001
7*      3      PITUUS(NAU),TYUT(NAUT3)      000001
8*      INTEGER EAUTO(NAU),SAUTO(NAU),EAURA(NUR),VAURA(NUR),      000001
9*      1      NRPOIS(NAU),NRGEN(NSK),URAT(NRE),SURA(NUR),FURA(NUR),      000001
10*      2      AUOMI(NAU),LVALU(IVAL),KONURA(NKON2),      000001
11*      3      KONFLP(NKON),VAIHE(LVT,IVAL),TILA(2,IVAL),HUOMAU(NAU)      000001
12*      4      JOND(NSK),JONOKA(NSK)      000001
13*      LOGICAL OHJAUS      000001
14*      COMMON/OS201/NREIT,NURA,NOM1,NSKAIS,NUKAIS,NAUTO      000001
15*      COMMON/OS202/XMIN,TKELL,C,LKMAUT,IT,XVAP      000001
16*      COMMON/OS203/OHJAUS,DUMHY(112)      000001
17*      COMMON/OS204/XSEIS,XLEV,XVALU,AALK,XKAANT,VKAANT      000001
18*      COMMON/OS207/KIRTUL(10),MIELKF(9),MRAJA,AIPIEN,AISUUR      000001
19*      COMMON/OS209/NKONFL      000001
20*      EQUIVALENCE (AU(NAU,5),PITUUS(NAU))      000001
21*      DATA NAUTO/NAU/IT/ITV/NOM1/NOM/      000001
22*      C TULOSTUSPARAMETRIEN JA TN-JAKAUTUMIEN LUKEMINEN      000001
23*      READ(5,584) TKIRJO      000001
24*      READ(5,584) TSIMU,KIRINP,KIRTUL,MIELKF      000001
25*      584 FORMAT(5X,F6.0,I1,10I5,9I2)      000024
26*      READ(5,588) MRAJA,AIPIEN,AISUUR      000024
27*      588 FORMAT(11,2F4,2)      000034
28*      READ(5,585) TNKFLT      000034
29*      READ(5,585) TNVALI      000044
30*      585 FORMAT(20F4,3)      000054
31*      C RISTEYKSEN DIMENSIOIDEN LUKEMINEN      000054
32*      READ(5,586) NURA,NSKAIS,NUKAIS,NREIT,NKONFL,LV,JAL      000054
33*      586 FORMAT(7I3)      000070
34*      READ(5,587) C,XVAP,XMIN,XSEIS,XVALU,XLEV,XKAANT,VKAANT,AALK      000070
35*      587 FORMAT(20F4,2)      000106
36*      IVAL=NSKAIS+JAL      000106
37*      NUR2=2*NURA      000111
38*      IF (KIRINP.EQ.1) WRITE(6,589)NSKAIS,NUKAIS,NURA,NREIT,NKONFL,IVALO      000114
39*      589 FORMAT('1 RISTEYKSEN DIMENSIOT')///      000133
40*      1' SISAANTULOKAISTOJEN LUKUMAARA',16//      000133
41*      2' ULOSMENOKAISTOJEN LUKUMAARA',16//      000133
42*      3' AJOURIEN LUKUMAARA',16//      000133
43*      4' AJOREITTIEN LUKUMAARA',16//      000133
44*      5' KONFLIKTIPISTEIDEN LUKUMAARA',16//      000133
45*      6' LIIKENNEVALOSUUNTIEN LUKUMAARA',16//      000133
46*      CALL INKONF(KONFLP,KONURA,NURA,NKONFL,NUR2,KIRINP)      000133
47*      CALL LUKU(URAT,PROS,XURA,VURA,VAIHE,TILA,SURA,FURA,IVALO,LV,EPIT,      000143
48*      1      LAMBDA,KIRINP)      000143
49*      C TAUUKKOMUUTTUJEN DIMENSIORAJOJEN TARKISTUS      000143
50*      IF (NURA.GT.NUR) CALL YLITYS('AJOURIEN')      000162
51*      IF (NSKAIS.GT.NSK) CALL YLITYS('SISAANTULOKAISTOJEN')      000172
52*      IF (NUKAIS.GT.NUK) CALL YLITYS('ULOSMENOKAISTOJEN')      000202
53*      IF (NREIT.GT.NRE) CALL YLITYS('AJOREITTIEN')      000212
54*      IF (NKONFL.GT.NKON) CALL YLITYS('KONFLIKTIPISTEIDEN')      000222
55*      IF (IVALO.GT.IVAL) CALL YLITYS('LIIKENNEVALOSUUNTIEN')      000232
56*      DO 510 I=1,NAUTO      000242
57*      510 NRPOIS(I)=1      000256
58*      X(NAUTO)=999.      000262
59*      V(NAUTO)=99.      000264
60*      XPYS(NAUTO)=99.      000266
61*      FLD(5,4,AUOMI(NAUTO))=1      000267
62*      DO 511 I=1,IT      000301
63*      511 A(NAUTO,I)=99.      000301
64*      KIERR=INT(TSIMU/C)      000303
65*      WRITE(6,579) TSIMU,C,KIERR      000314
66*      579 FORMAT('11 SIMULOIDAAN RISTEYSLIIKENNETTA REAALIAJASSA',F6.0,      000324
67*      1' SEKUNTIA,1/1 AIKA=ASKELENA ONI,F5.2,1' SEKUNTIA,1/      000324
68*      2' KAIKKIAAN SIMULOIDAAN SIISI,16,1' KIERROSTA,1/))      000324
69*      IK=0      000324
70*      500 IK=IK+1      000326
71*      TTULOS=KIRTUL(IK)      000332
72*      IF (TTULOS.LE.0) TTULOS=TSIMU      000334
73*      DO 516 I=1,NAUTO      000343
74*      516 HUOMAU(I)=1      000347

```



```

75*      CALL PAIVIT(X,V,A,SAUTO,EAUTO,FAURA,VAURA,SURA,TPYS,XPYS,AUOMI,      000351
76*      1      XURA,EURA,XVALI,URAT,HUOMAU,PITUUS,TYOT,NAUT3)      000351
77*      CALL POISTU(X,NSKAIS,NUKAIS,NAUTO,NURA,EAURA,XURA,SAUTO,      000376
78*      1      EAUTO,AUOMI,NRPOIS,URAT, NREIT,TKELL,AU,NOM,TPYS,VAURA)      000376
79*      IF (TKELL,LT,TKIRJO) GO TO 400      000422
80*      DELETE 400,DPAI,LPAT      000430
81*      WRITE(6,595)      000430
82*      595 FORMAT(99H PAIVITYS: X V A      TPYS      000430
83*      1SAUTO EAUTO EAURA VAURA      )      000430
84*      DO 550 I=1,NAUTO      000430
85*      550 WRITE(6,590) X(I),V(I),(A(I,J),J=1,IT),TPYS(I),SAUTO(I),      000430
86*      1EAUTO(I),EAURA(I),VAURA(I),SURA(I),EURA(I)      000430
87*      590 FORMAT(1X,6F6,2,5I5,2A6)      000430
88*      WRITE(6,593) TKELL,LKMAUT      000430
89*      400 CONTINUE      000430
90*      CALL URVAIH(NAUTO,X,AUOMI,EAUTO,SAUTO,NURA,EAURA,VAURA,XURA,      000430
91*      1      SURA,EURA,NREIT,URAT,NSKAIS,VURA,AU,NOMI,XVALI)      000430
92*      IF (TKELL,LT,TKIRJO) GO TO 410      000453
93*      DELETE 410,DURV,LURV      000461
94*      WRITE(6,594)      000461
95*      594 FORMAT(12H URVAIHTO: X)      000461
96*      DO 553 I=1,NAUTO      000461
97*      IS=FLD(16,6,AUOMI(I))      000461
98*      553 WRITE(6,598) X(I),IS,EAURA(I),VAURA(I)      000461
99*      598 FORMAT(F16,3,110,2I6)      000461
100*      410 CONTINUE      000461
101*      CALL GENERO(X,V,A,AU,AUOMI,VAURA,NRGEN,URAT,XURA,LAMBDA,TGEN,XVALI      000461
102*      1      ,EAURA,SAUTO,EAUTO,PROS,HUOMAU,NRPOIS,EPIT,EURA,ALFA)      000461
103*      IF (TKELL,LT,TKIRJO) GO TO 420      000507
104*      DELETE 420,DGEN,LGEN      000515
105*      WRITE(6,596)      000515
106*      596 FORMAT(108H GENERO: X V A      XVALI VAURA      000515
107*      1NRGEN EAUTO XURA LVALO TGEN EAURA SAUTO AU      )      000515
108*      DO 551 I=1,NAUTO      000515
109*      551 WRITE(6,591) X(I),V(I),(A(I,J),J=1,IT),XVALI(I),VAURA(I),NRGEN(I),      000515
110*      1EAUTO(I),XURA(I),LVALO(I),TGEN(I),EAURA(I),SAUTO(I),(AU(I,J),J=1,      000515
111*      2NOM),SURA(I),EURA(I)      000515
112*      591 FORMAT(1X,6F6,2,3I6,F4,0,14, F6,2,2I4,5F6,2,2A6)      000515
113*      WRITE(6,593) TKELL,LKMAUT      000515
114*      420 CONTINUE      000515
115*      IF (,NOT,OHJAUS) GO TO 566      000515
116*      CALL LVALOT(LVALO,C,TILA,VAIHE,LV,LVALO,NSKAIS)      000516
117*      566 CALL JONOT(X,V,EAURA,JOVO,JONOKA,SAUTO)      000530
118*      CALL KIIHTY(X,V,A,ALFA,EAUTO,SAUTO,VAURA,EAURA,XPYS,AU,AUOMI,      000537
119*      1      SURA,LVALO,XURA,XVALI,TKELL,TVALI,EURA,      000537
120*      2      KONFLP,KONURA,NKONFL,NUR2,URAT,OHJAUS)      000537
121*      DELETE 430,DKII,LKII      000571
122*      IF (TKELL,LT,TKIRJO) GO TO 430      000571
123*      WRITE(6,597)      000571
124*      597 FORMAT(1 KIIHTY: X V A      URA NR EA      000571
125*      1UTO SAUTO VAURA EAURA XPYS TPYS YHT KNF XVALI TGEN LVALO)      000571
126*      MINJAM=IN2(LKMAUT,99)      000571
127*      DO 552 I=1,MINJA      000571
128*      IF (FLD(9,1,AUOMI(I)),F0,1,AND,I,GT,NSKAIS) GO TO 552      000571
129*      IKEL=FLD(2,2,AUOMI(I))      000571
130*      MERKKI=FLD(5,4,AUOMI(I))      000571
131*      IU =FLD(16,6,AUOMI(I))      000571
132*      IKO =FLD(22,1,AUOMI(I))      000571
133*      WRITE(6,592) X(I),V(I),(A(I,J),J=1,IT),IU,I ,EAUTO(I),SAUTO(I),      000571
134*      1VAURA(I),EAURA(I),XPYS(I),TPYS(I),MERKKI ,IKEL,IKO,XVALI(I),      000571
135*      2TGEN(I),LVALO(I),HUOMAU(I)      000571
136*      552 CONTINUE      000571
137*      592 FORMAT(1X,5F6,2,12X,6I6,2F6,2,3I4,2F6,2,14,1X,A6)      000571
138*      100 WRITE(6,593) TKELL,LKMAUT      000571
139*      430 CONTINUE      000571
140*      593 FORMAT(1X,F6,2,15)      000571
141*      IF (TKELL,LT,TAYITO) GO TO 515      000571
142*      IF (TKELL,GT,TSIMU) CALL KIRJAA(599,TKELL)      000576
143*      IF (TKELL,GT,TTULOS) CALL KIRJAA(590,TKELL)      000607
144*      GO TO 515      000620
145*      599 STOP      000622
146*      SUBROUTINE YLITYS(MAPU)      000625
147*      INTEGER MAPU(4)      000625
148*      WRITE(6,578) MAPU      000625
149*      578 FORMAT(11 VIHETILANNE: 1,4A6,1 LUKUMAARA LIIAN SUURI)      000640
150*      STOP      000640
151*      END      000661

```



```

1*      SUBROUTINE LUKU(URAT,PROS,XURA,VURA,VAIME,TILA,SURA,EURA,IVALO,LV,      000077
2*      EPIT,LAMBDA,KIRINP)      000077
3*      INTEGER URAT(NREIT),IAPU(6),SURA(NURA),EURA(NURA),IA(3)      000077
4*      REAL PROS(NREIT),XURA(NURA),VURA(NURA),EPIT(12,NSKAIS),      000077
5*      LAMBDA(5,NSKAIS)      000077
6*      INTEGER VAIME(LV,IVALO),TILA(2,IVALO)      000077
7*      LOGICAL OHJAUS      000077
8*      COMMON/OS201/NREIT,NURA,DUMMY,NSKAIS,NUKAIS,DUMMY1      000077
9*      COMMON/OS202/XMIN,TKELL,C,LKMAUT, IT,XVAP      000077
10*     COMMON/OS203/OHJAUS,SKALA(8,2),TNJAL(4,20),DUMMY2(16)      000077
11*     C REITTITIETOJEN LUKEMINEN      000077
12*     DO 1020 J=1,NREIT      000077
13*     READ(5,1090,END=1060) IAPU,PROS(J)      000077
14*     1090 FORMAT(10X,6I2,2X,F3,2)      000112
15*     C REITTIKOHTAISTEN LIIKENNEMAARIEN TARKISTUS      000112
16*     IF (J,EQ,1) GO TO 1005      000112
17*     IF (IAPU(1),EQ,FLD(0,6,URAT(J=1))) GO TO 1005      000116
18*     IF (ABS(SUM-1.),GT,1E=5) WRITE(6,1006) J      000122
19*     1006 FORMAT(' VIRHE: REITTIAI,13, ENNEN LUETUT LIIKENNEMÄÄRÄDATAT VIR      000137
20*     MEELLISIA, SUMMA EI OLLUT = 1 (FRMT 1006)')      000137
21*     SUM=0      000137
22*     1005 SUM=SUM+PROS(J)      000141
23*     K=0      000143
24*     1010 FLD(6*K,6,URAT(J))=IAPU(K+1)      000145
25*     K=K+1      000160
26*     IF (IAPU(K+1),EQ,0,OR,K,GT,6) GO TO 1020      000164
27*     GO TO 1010      000176
28*     1020 CONTINUE      000213
29*     C URATITIETOJEN LUKEMINEN      000213
30*     DO 1030 I=1,NURA      000213
31*     READ(5,1091) IAPU(6),IAPU(1),IAPU(2),XURA(I),VURA(I)      000225
32*     1091 FORMAT(5X,12,3X,2I2,1X,F4,1,3X,F3,1)      000237
33*     IF (I,GT,NSKAIS) GO TO 1029      000237
34*     READ(30,1092)(LAMBDA(J,I),J=1,5),(EPIT(J,I),J=1,12)      000244
35*     1092 FORMAT(25X,4F4,1,F3,2,3(F4,2,F4,1,F2,1,F2,0))      000265
36*     IF (ABS(EPIT(1,I)+EPIT(5,I)+EPIT(9,I)=1.),GT,1E=6) GO TO 1061      000265
37*     1029 FLD(0,6,SURA(I))=IAPU(1)      000277
38*     1030 FLD(6,6,SURA(I))=IAPU(2)      000300
39*     IF (IAPU(6),NE,NURA) WRITE(6,1093)      000307
40*     1093 FORMAT(' URADATOJEN LUKUMAARA VIRHEELLINEN (FRMT 1093)')      000320
41*     DO 1036 J=0,1      000320
42*     DO 1035 I=NSKAIS,NURA      000342
43*     IND=FLD(6*J,6,SURA(I))      000342
44*     IF (IND,EQ,0) GO TO 1035      000346
45*     FLD(6*J,6,EURA(IND))=I      000350
46*     1035 CONTINUE      000366
47*     1036 CONTINUE      000366
48*     C LIIKENNEVALO=OHJAUSDATOJEN LUKEMINEN      000366
49*     READ(5,1080) OHJAUS,KIERTO      000366
50*     1080 FORMAT(L1,I3)      000375
51*     IF (.NOT.OHJAUS) GO TO 1049      000375
52*     C LUETAAN KULLEKIN SISÄÄNTULOJAISTALLE JA JALANKULKUSUOJATIELLE ALKUTILAMATRIISI      000375
53*     C JA VALOJEN KESTOAJAT      000375
54*     DO 1040 I=1,IVALO      000412
55*     READ(5,1081) (TILA(J,I),J=1,2),(VAIME(J,I),J=1,LV)      000412
56*     1081 FORMAT(10X,11,1X,12,1X,32I2)      000427
57*     TILA(2,I)=TILA(2,I)/C      000427
58*     IF (I,LE,NSKAIS) GO TO 1046      000442
59*     SKALA(I=NSKAIS,1)=VAIME(1,I)/10,      000447
60*     SKALA(I=NSKAIS,2)=VAIME(3,I)/10,      000456
61*     1046 ISUM=0      000464
62*     DO 1045 J=1,LV      000466
63*     1045 ISUM=ISUM+VAIME(J,I)      000466
64*     IF (ISUM,NE,KIERTO) WRITE(6,1096) I      000471
65*     1096 FORMAT(' VIRHETILANNE: 1,13,1IN VALOVAIHEIDEN SUMMA EI TASHANNYT      000512
66*     KIERTOAJAN KANSSA (FRMT 1096)')      000512
67*     1040 CONTINUE      000512
68*     DO 1050 I=1,4      000512
69*     1050 READ(5,1082) (TNJAL(I,J),J=1,20)      000512
70*     1082 FORMAT(20F4,3)      000527

```



```

1*      SUBROUTINE INKONF(KONFLP,KONURA,NURA,NKONFL,NUR2,KIRINP)
2*      INTEGER KONFLP(NKONFL),KONURA(NUR2)
3*      COMMON/OS206/SEKTOR(8)
4*      LISA=NKONFL+1
5*      READ(5,1179) SEKTOR
6*      1179 FORMAT(5X,8F4,0)
7*      I=0
8*      1199 I=I+1
9*      READ(5,1198,END=1194) ITESTI,IA,IB,IC,ID,IE,IF
10*     1198 FORMAT(5X,12,3X,2I2,1X,2I3,2I4)
11*      FLD(0,6,KONFLP(I))=IA
12*      FLD(6,6,KONFLP(I))=IB
13*      FLD(12,7,KONFLP(I))=IC
14*      FLD(19,7,KONFLP(I))=ID
15*      J=0
16*      1191 J=J+1
17*      IF (IE,GE,SEKTOR(J)) GO TO 1189
18*      IF (J,LT,8) GO TO 1191
19*     1189 FLD(26,3,KONFLP(I))=J-1
20*      FLD(29,3,KONFLP(I))=8-J
21*      FLD(32,4,KONFLP(I))=IF
22*      IND=0
23*     1198 IND=IND+1
24*      IAPU=I
25*      C ETSITAAN JARJESTETTY PAIKKA, JOHON KONFLIKTIPISTE KUULUU
26*      J=1
27*     1195 J=J+1
28*      IAPV=FLD(7*J,7,KONURA(IA))
29*      IF (IAPV,EQ,0) GO TO 1183
30*      IF (J,EQ,4) GO TO 1192
31*      K=1
32*     1180 K=K+1
33*      IF (K,GT,2) GO TO 1170
34*      IF (IA,NE,FLD(6*K,6,KONFLP(IAPV)),AND,
35*      1 IATALL,NE,FLD(6*K,6,KONFLP(IAPV))) GO TO 1180
36*      IF (IC,LT,FLD(12+7*K,7,KONFLP(IAPV))) GO TO 1184
37*      GO TO 1195
38*      C PAIKKA LOYTYY VASTA JATKOKETJUSTA
39*     1192 IF (IAPV,LE,NKONFL) GO TO 1186
40*      IATALL=IA
41*      IA=FLD(28,7,KONURA(IA))
42*      J=1
43*      GO TO 1195
44*      C KONFLIKTIPISTE SIJOITETAAN TYHJAAN PAIKKAAN KETJUN LOPPUUN
45*     1183 FLD(7*J,7,KONURA(IA))=IAPU
46*      GO TO 1188
47*      C KONFLIKTIPISTE SIJOITETAAN KETJUN KESKELLE
48*     1184 FLD(7*J,7,KONURA(IA))=IAPU
49*      IAPU=IAPV
50*      J=J+1
51*      C SIIRRETAAN KETJUN LOPPUOSA ASKELEN VERRAN TAAKSEPAIN
52*     1185 IAPV=FLD(7*J,7,KONURA(IA))
53*      IF (IAPV,EQ,0) GO TO 1183
54*      IF (J,LT,4) GO TO 1184
55*      IF (IAPV,LE,NKONFL) GO TO 1187
56*      IA=FLD(28,7,KONURA(IA))
57*      J=0
58*      GO TO 1184
59*      C MUODOSTETAAN URALLE IA JATKO=KONFLIKTIPISTEKETJU
60*     1186 FLD(28,7,KONURA(IA))=LISA
61*      IATALL=IA
62*      IA=LISA
63*      FLD(0,7,KONURA(IA))=IAPV
64*      LISA=LISA+1
65*      J=1
66*      GO TO 1195
67*      C MUODOSTETAAN JATKOKETJU JA SIIRRETAAN ENTISEN KETJUN LOPPUPAA SIIHEN
68*     1187 FLD(28,7,KONURA(IA))=LISA
69*      FLD(0,7,KONURA(LISA=NKONFL+NURA))=IAPU
70*      FLD(7,7,KONURA(LISA=NKONFL+NURA))=IAPV
71*      LISA=LISA+1
72*      C OTETAAN KONFLIKTIPISTEEN TOINEN AJOURA KASITELTAVAKSI
73*     1188 IF (IATALL,GT,0) IA=IATALL
74*      IF (IATALL,GT,0) IATALL=0
75*      IAPU=IA
76*      IA=IB
77*      IB=IAPU
78*      IAPU=IC
79*      IC=ID
80*      ID=IAPU
81*      IF (IND,LT,2,AND,IA,LE,NURA) GO TO 1190
82*      IF (I,LT,NKONFL) GO TO 1199
83*     1194 IF (ITESTI,NE,1) WRITE(6,1177)
84*     1177 FORMAT(1 VIRHETILANNE KONFLIKTIPISTEDATOJEN LUKUMAARASSA VIRHE (
85*     1FRMT 1177))

```

```

000010
000010
000010
000010
000013
000023
000023
000025
000027
000046
000046
000050
000052
000060
000066
000070
000072
000102
000107
000120
000127
000134
000136
000140
000140
000142
000145
000147
000156
000160
000164
000167
000171
000177
000177
000220
000232
000232
000235
000241
000243
000251
000253
000253
000255
000271
000271
000273
000307
000311
000311
000314
000321
000323
000330
000334
000340
000341
000341
000343
000352
000354
000356
000365
000367
000371
000371
000373
000402
000413
000421
000421
000425
000432
000437
000441
000443
000444
000446
000450
000451
000466
000474
000504
000504

```



```

86* C KONFLIKTIPISTEKOHTAISTEN SYÖTTÖTIEDOTEN KIRJOITTAMINEN 000504
87* IF (KIRINP,NE,1) GO TO 115H 000504
88* WRITE(6,1181) 000512
89* 1181 FORMAT('1 KONFLIKTIPISTEKOHTAISET SYÖTTÖTIEDOT'/// 000525
90* 1' KONFLIKTI= LEIKKAAVAT K=PISTEEN ETAISYYS URIEN KON 000525
91* 2TAUSKULMAT NAKOESTEEN ETAISYYS/ 000525
92* 3' PISTE AJOURAT EDELL,URAN JALK,URAN JALK,URA 000525
93* 4N EDELL,URAN KONFLIKTIPISTEESTA/ 000525
94* 533X,2('ALUSTA '),2('TULOSEKTORI '),'1 MITATTUNA/ 000525
95* 657X,'EDELL,URAN JALK,URAN URAA PITKINI/ 000525
96* 757X,'SUHTEEN SUHTEEN') 000525
97* DO 1197 I=1,NKONFL 000525
98* IA=FLD(0,6,KONFLP(I)) 000525
99* IB=FLD(6,6,KONFLP(I)) 000527
100* IC=FLD(12,7,KONFLP(I)) 000531
101* ID=FLD(19,7,KONFLP(I)) 000535
102* IE=FLD(26,3,KONFLP(I)) 000541
103* IF=FLD(29,3,KONFLP(I)) 000545
104* IG=FLD(32,4,KONFLP(I)) 000551
105* 1197 WRITE(6,1196) I,IA,IB,IC,ID,IE,IF,IG 000555
106* 1196 FORMAT(5X,13,12X,12,' = ',12,110,112,3113) 000574
107* WRITE(6,1182) 000574
108* 1182 FORMAT('1 AJOURAKOHTAISET KONFLIKTIPISTELUETTELOT'/// 000605
109* 2' AJOURA KONFLIKTIPISTEET'//) 000605
110* DO 1193 I=1,NUR2 000605
111* IF (KONURA(I),EQ,0,AND,I,GT,NURA) GO TO 1150 000620
112* IA=FLD(0,7,KONURA(I)) 000623
113* IB=FLD(7,7,KONURA(I)) 000627
114* IC=FLD(14,7,KONURA(I)) 000633
115* ID=FLD(21,7,KONURA(I)) 000637
116* IE=FLD(28,7,KONURA(I)) 000643
117* WRITE(6,1175) I,IA,IB,IC,ID,IE 000656
118* 1175 FORMAT(18,6X,613) 000656
119* IF (I,GT,NURA) WRITE(6,1176) 000674
120* 1176 FORMAT(1H+,34X,'JATKOLUETTELO') 000674
121* 1193 CONTINUE 000677
122* 1150 RETURN 000704
123* 1170 WRITE(6,1171) ! 000704
124* 1171 FORMAT('1 VIRHETILANNE: ',13,'INNEEN KONFLIKTIPISTEEN SIJOITTAMINEN 000704
125* 1EI ONNISTUNUT (FRMT 1171)') 000704
126* STOP 000745
127* END

```



```

1*      SUBROUTINE PAIVIT(X,V,A,SAUTO,EAUTO,EAURA,VAURA,SURA,TPYS,XPYS,
2*      1 AUOMI,XURA,EURA,XVALI,URAT,HUOMAU,PITUUS,TYOT,NAUT3)
3*      COMMON/OS201/NREIT,NURA,DUMMY,NSKAIS,NUKAIS,NAUTO
4*      COMMON/OS202/XMIN,TKELL,C,DUMMY1,IT,DUMMY3
5*      REAL X(NAUTO),V(NAUTO),A(NAUTO,IT),TPYS(NAUTO),XPYS(NAUTO),
6*      1 XURA(NAUTO),XVALI(NAUTO),PITUUS(NAUTO),TYOT(NAUT3)
7*      INTEGER SAUTO(NAUTO),EAURA(NURA),VAURA(NURA),SURA(NURA),EURA(NURA)
8*      1 ,EAUTO(NAUTO),AUOMI(NAUTO),URAT(NREIT),HUOMAU(NAUTO)
9*      LOGICAL KESKEN
10*     DEFINE XTALL(N)=TYOT(N)
11*     DEFINE VTALL(N)=TYOT(N+NAUTO)
12*     DEFINE ATALL(N)=TYOT(N+2*NAUTO)
13*     DEFINE ETAIS(N)=XVALI(N)+X(KAPU)=XTALL(KAPU)=(X(N)-XTALL(N))
14*     KESKEN=.FALSE.
15*     TKELL=TKELL+C
16*     DO 2001 IK=1,NUKAIS
17*     1 IK=NSKAIS
18*     ISUOR=I
19*     C ONKO URALLA AJONEUVOA)
20*     2003 IF (EAURA(I),LE,0) GO TO 2070
21*     N=EAURA(I)
22*     C MAETAAN URALLA KAIKKI AUTOT JA TALLETETAAN NIITTEN LIIKKEET
23*     2025 XTALL(N)=X(N)
24*     VTALL(N)=V(N)
25*     IF (FLD(5,4,AUOMI(N)),NE,3) GO TO 2024
26*     DO 2023 J=1,IT
27*     2023 A(N,J)=A(N,IT)
28*     2024 ATALL(N)=A(N,1)
29*     IF (IT,LT,2) GO TO 2031
30*     DO 2030 J=2,IT,1
31*     2030 A(N,J)=A(N,J)
32*     2031 IF (I,GT,NURA,OR,N,EQ,0) GO TO 2090
33*     IF (N,EQ,VAURA(I)) GO TO 2020
34*     N=SAUTO(N)
35*     GO TO 2025
36*     C LASKETAAN UUSI NOPEUS JA PAIKKA
37*     2020 N=EAURA(I)
38*     2021 V(N)=VTALL(N)+ATALL(N)*C
39*     X(N)=XTALL(N)+VTALL(N)*C+0,5*ATALL(N)*C*C
40*     C KASITELTAVA AUTO PYSÄHTYÄ JUURI NYT
41*     IF (,NOT,(VTALL(N),GT,1E=2,AND,V(N),LE,1E=2)) GO TO 2040
42*     X(N)=XTALL(N)+VTALL(N)*(-VTALL(N)/ATALL(N))+0,5*ATALL(N)*(-VTALL(N)
43*     1)/ATALL(N))*2
44*     2022 V(N)=0
45*     DO 2035 J=1,IT
46*     2035 A(N,J)=0
47*     FLD(5,4,AUOMI(N))=0
48*     2040 KAPU=EAUTO(N)
49*     IF (KAPU,LE,0) GO TO 2080
50*     C LASKETAAN MINIMIVALIMATKA JA TESTATAAN SITA
51*     2045 IF (ETAIS(N),GE,0,8*PITUUS(KAPU)) GO TO 2085
52*     C LASKETAAN UUSI LIIKE, JOKA EI VIE AUTOA MINIMIVALIMATKAA LAHEMMAKSI EDELTAJAN
53*     HUOMAU(N)=IPERAANI
54*     X(N)=XTALL(N)
55*     ATALL(N)=2*(ETAIS(N)+0,001 -0,8*PITUUS(KAPU)=VTALL(N)*C)/(C*C)
56*     V(N)=VTALL(N)+ATALL(N)*C
57*     X(N)=XTALL(N)+VTALL(N)*C+0,5*ATALL(N)*C*C
58*     A(N,1)=ATALL(N)
59*     IF (V(N),LE,1E=2) GO TO 2022
60*     GO TO 2085
61*     2080 XVALI(N)=9999.
62*     2085 IF (KAPU,GT,0) XVALI(N)=ETAIS(N)
63*     C JOS KASITELTAVA AUTO ON PYSÄHTYNYT, KASVATETAAN SEISONTA-AIKAA
64*     IF (VTALL(N),LE,1E=2) TPYS(N)=TPYS(N)+C
65*     C JOS KASITELTAVA AUTO ON LAHDOSSE LIIKKEELLE, NOLLATAAN PYSÄHTYMISPAIKAN ARVO
66*     2087 IF (FLD(5,4,AUOMI(N)),NE,3,AND,FLD(5,4,AUOMI(N)),NE,0) XPYS(N)=0
67*     C ONKO KASITTELEMATTOMIA AUTOJA AJOURALLA)
68*     IF (N,EQ,VAURA(I)) GO TO 2070
69*     N=SAUTO(N)
70*     GOTO 2021
71*     C ONKO KASITTELEMATTOMIA AJOURIA JALJELLA TALLA AIKA-ASKELELLA)
72*     C JOS URARISTEYKSEEN TULEE KÄÄNTYVÄ URA, TUTKITAAN ENSIN SUORA JA SITTEN KÄÄREVA
73*     2070 IF (KESKEN) GO TO 2075
74*     IF (ISUOR,LE,NSKAIS) GO TO 2001
75*     IKAANT=FLD(6,6,SURA(ISUOR))
76*     ISUOR=FLD(0,6,SURA(ISUOR))
77*     IF (IKAANT,EQ,0) GO TO 2076
78*     IF (FLD(0,6,EURA(IKAANT)),NE,0) GO TO 2076
79*     IF (ISUOR,EQ,0) GO TO 2075
80*     I=ISUOR
81*     KESKEN=.TRUE.
82*     GO TO 2003
83*     2075 IF (I,EQ,IKAANT) GO TO 2074
84*     I=IKAANT
85*     GO TO 2003
86*     2074 IF (FLD(0,6,SURA(IKAANT)),GT,0) I=FLD(0,6,SURA(IKAANT))
87*     KESKEN=.FALSE.
88*     IF (I,EQ,IKAANT) GO TO 2070
89*     GO TO 2003
90*     2076 I=ISUOR
91*     IF (I,EQ,0) GO TO 2001
92*     GO TO 2003
93*     2001 CONTINUE
94*     RETURN
95*     2090 WRITE(6,2091) I,N
96*     2091 FORMAT(' VIRHETILANNE: ',I4,' INNEN URAN KASITTELYSSA VIRHE, AUTON
97*     1 ID=KODI1',I4,' TAUUKUOIDEN EAURA JA VAURA DUMPPAUS: ',I)
98*     WRITE(6,2092) EAURA,VAURA
99*     2092 FORMAT('
100*     STOP
101*     END

```



```

1*      SUBROUTINE URVAITH(AUTO,X,AUOMI,EAUTO,SAUTO,NURA,EAURA,VAURA,XURA,
2*      1      SUKA,EURA,NREIT,URAT,NSKAIT,VURA,AU,NOMI,XVALI)
3*      REAL X(AUTO),XURA(NURA),VURA(NURA),AU(AUTO,NOMI),XVALI(AUTO)
4*      INTEGER AUOMI(AUTO),EAUTO(AUTO),SAUTO(AUTO),FAURA(NURA),
5*      1      VAURA(NURA),EURA(NURA),URAT(NREIT),SURA(NURA)
6*      DO 2270 IK=1,NURA
7*      1=IK
8*      2203 N=EAURA(I)
9*      IF (N.LE.0) GO TO 2270
10*     C ONKO URALLA ENSIMMAISENA OLEVA AUTO AJANUT UUELLE AJOURALLE)
11*     IF (X(N)=XURA(I),LT.0) GO TO 2270
12*     X(N)=X(N)-XURA(I)
13*     C MAETAAN URALLA ENSIMMAISENA OLEVALLE AUTOLLE SEURA-AVAKSI TULEVA AJOURA
14*     2250 NRREIT=FLD(10,6,AUOMI(N))
15*     DO 2210 J=0,6
16*     IF (FLD( 6,J,6,URAT(NRREIT))=1,EQ.0) GO TO 2220
17*     IF (J,GE.5) WRITE(6,2290)
18*     2290 FORMAT(' VIRHETILANNE: KASITELTAVAN AUTON URATIETO EI TASHAX AUTO
19*     1N REITIN URATIEDON KANSSA (FRMT 2293)')
20*     2210 CONTINUE
21*     2220 NRURA=FLD( 6*(J+1),6,URAT(NRREIT))
22*     C AUTON TAVOITENOPEUS UUELLE URALLA
23*     AU(N,2)=VURA(NRURA)
24*     C ONKO AUTO AJANUT REITINSIA LOPPUUN)
25*     IF (NRURA,LE.0) GO TO 2270
26*     FLD(16,6,AUOMI(N))=NRURA
27*     C JOS AUTO KAANTYY UUDEN URAN JALKEEN, MERKITAAN SE MUUTTUJAAN AUOMI
28*     IF (J,GT.3) GO TO 2225
29*     IF (FLD(0,6,EURA(NRURA)),EQ,FLD(6*(J+2),6,URAT(NRREIT)))
30*     1      FLD(0,2,AUOMI(N))=0
31*     IF (FLD(6,6,EURA(NRURA)),EQ,FLD(6*(J+2),6,URAT(NRREIT)))
32*     1      FLD(0,2,AUOMI(N))=1
33*     C ETENEKKO ENSIMMAINEN AUTO SUORAAN VAI KAANTYEN UUTEEN URAAN)
34*     2225 IF (FLD(0,6,EURA(I)),EQ,NRURA) GO TO 2230
35*     IF (FLD(6,6,EURA(I)),EQ,NRURA) GO TO 2240
36*     WRITE(6,2209) NRURA
37*     2209 FORMAT(' VIRHETILANNE: URALUETTELOON MUKAAN SEURA-AVANA EI VOI OLLA
38*     1URA 1,16,(FRMT 2209)')
39*     STOP
40*     C AUTO ETENEE AJOURAA PITKIN SUORAAN ETEENPAIN
41*     2230 VAURA(NRURA)=N
42*     EAURA(I)=SAUTO(N)
43*     IF (EAURA(NRURA),EQ.0) EAURA(NRURA)=N
44*     KAPU1=SAUTO(N)
45*     IF (KAPU1,GT.0,AND,FLD(16,6,AUOMI(KAPU1)),EQ,1) GO TO 2270
46*     EAURA(I)=0
47*     VAURA(I)=0
48*     GO TO 2270
49*     C AUTO KAANTYY JA LIITTYY TOISEEN LIIKENNEVIRTAAN
50*     C VANHAN SUUNTAISEN LIIKENTEEN UUELLEENKETJUTUS
51*     2240 KAPU1=EAUTO(N)
52*     KAPU2=SAUTO(N)
53*     IF (KAPU2,GT.0) XVALI(KAPU2)=XVALI(N)+XVALI(KAPU2)
54*     IF (KAPU2,GT.0) EAUTO(KAPU2)=KAPU1
55*     IF (KAPU1,GT.0) SAUTO(KAPU1)=KAPU2
56*     EAURA(I)=SAUTO(N)
57*     IF (KAPU2,GT.0,AND,FLD(16,6,AUOMI(KAPU2)),EQ,1) GO TO 2239
58*     EAURA(I)=0
59*     VAURA(I)=0
60*     C UUDEN SUUNTAISEN LIIKENTEEN UUELLEEN KETJUTUS
61*     2239 IF (VAURA(NRURA),NE.0) GO TO 2243
62*     VAURA(NRURA)=N
63*     EAURA(NRURA)=N
64*     IAPU=NRURA
65*     XVALI(N)=XURA(IAPU)=X(N)
66*     C ETSITAAN EDESTAPAIN EAUTOA
67*     2242 IAPU=FLD(0,6,EURA(IAPU))
68*     IF (IAPU,EQ.0) GO TO 2241
69*     XVALI(N)=XVALI(N)+XURA(IAPU)
70*     IF (VAURA(IAPU),EQ.0) GO TO 2242
71*     KAPU1=VAURA(IAPU)
72*     XVALI(N)=XVALI(N)+(XURA(IAPU)=X(KAPU1))
73*     GO TO 2244
74*     C ETSITAAN TAKAAPAIN SAUTOA
75*     2241 IAPU=NRURA
76*     XVALI(N)=9999
77*     2246 IAPU=FLD(0,6,SURA(IAPU))
78*     IF (IAPU,EQ.0) GO TO 2260
79*     IF (EAURA(IAPU),EQ.0) GO TO 2246
80*     KAPU2=EAURA(IAPU)
81*     GO TO 2245
82*     C UUELTA AJOURALTA LOYTYI AUTO
83*     2243 KAPU1=VAURA(NRURA)
84*     XVALI(N)=X(KAPU1)=X(N)
85*     VAURA(NRURA)=N
86*     IF (EAURA(NRURA),EQ.0) EAURA(NRURA)=N
87*     C UUESTA AJOSUUNNASTA AUTO LOYTYI VASTA KAUKAISelta AJOURALTA
88*     2244 KAPU2=SAUTO(KAPU1)
89*     SAUTO(KAPU1)=N
90*     C VAIN TAKAAPAIN TULEVA AUTO LOYTYI UUESTA AJOSUUNNASTA
91*     2245 IF (KAPU2,GT.0) EAUTO(KAPU2)=N
92*     SAUTO(N)=KAPU2
93*     EAUTO(N)=KAPU1
94*     GO TO 2270
95*     C UUESTA AJOSUUNNASSA EI OLE ENNESTAAN AUTOJA
96*     2260 SAUTO(N)=0
97*     EAUTO(N)=0
98*     2270 CONTINUE
99*     RETURN
100*    END

```



```

1*      SUBROUTINE GENERO(X,V,A,AU,AUOMI,VAURA,NRGEN,URAT,XURA,LAMBDA,TGEN      000120
2*      1,XVALI,EAURA,SAUTO,EAUTO,PROS,HUOMAU,NRPOIS,EPIT,EURA,ALFA)      000120
3*      COMMON/OS201/NREIT,NURA,NOMI,NSKAIS,NUKAIS,NAUTO      000120
4*      COMMON/OS202/DUMMYS,TKELL,C,LKMAUT,IT,DUMMYS      000120
5*      COMMON/OS205/LUOKKA(5)      000120
6*      RZAL X( NAUTO ),V(NAUTO ),A(NAUTO,IT),AU(NAUTO,NOMI),XURA(NURA),      000120
7*      1 LAMBDA(5,NSKAIS),TGEN(NSKAIS),PROS(NREIT),XVALI(NAUTO),      000120
8*      2 EPIT(12,NSKAIS),ALFA(NAUTO),PAR(7)      000120
9*      INTEGER AUOMI(NAUTO),NRGEN(NSKAIS),URAT(NREIT),SAUTO(NAUTO),      000120
10*      1 VAURA(NURA),EAURA(NURA),EAUTO(NAUTO),HUOMAU(NAUTO),      000120
11*      2 NRPOIS(NAUTO),EURA(NURA)      000120
12*      2408 FORMAT('1 VIRHETILANNEI KASITELTAVANA OLEVA URANUMERO JA AUTON URA      000120
13*      1TIIETO ERILAISIA (FRMT 2408)')      000120
14*      C MAETAAN NE SISAANTULOKAISTAT,JOIHIN ON GENEROITU AUTOT      000120
15*      DO 2409 NRURA=1,NSKAIS      000120
16*      IF (TGEN(NRURA),GT,TKELL) GO TO 2409      000134
17*      C SOVITETAAN AUTO AJOURALLE      000134
18*      N=NRGEN(NRURA)      000141
19*      IF (N,LE,0) GO TO 2419      000143
20*      NRREIT=FLD(10,6,AUOMI(N))      000146
21*      NRURI=FLD(0,6,URAT(NRREIT))      000160
22*      IF (NRURI,NE,NRURA) WRITE(6,2408)      000163
23*      KAPU=VAURA(NRURA)      000170
24*      XVALI(N)=0      000176
25*      IF (KAPU,GT,0) GO TO 2406      000200
26*      C LÖYTYKÖ EDELTAJAAJOURILTA AUTOA)      000200
27*      IAPU=NRURA      000203
28*      2405 XVALI(N)=XVALI(N)+XURA(IAPU)      000206
29*      IAPU=FLD(0,6,EURA(IAPU))      000216
30*      IF (IAPU,LE,NSKAIS+NUKAIS,AND,IAPU,NE,FLD(0,6,EURA(NRURA)))      000220
31*      1 GO TO 2410      000220
32*      IF (VAURA(IAPU),EQ,0) GO TO 2405      000236
33*      KAPU=VAURA(IAPU)      000242
34*      C EDELTAJAAUTO ON KAPU      000242
35*      2406 XVALI(N)=XVALI(N)+X(KAPU)-X(N)      000245
36*      IF (XVALI(N),GT,AU(KAPU,5)) GO TO 2411      000264
37*      HUOMAU(N)=ITAYSI 1      000270
38*      C UUSI TULOAIKA LASKETAAN AIKA=ASKELEEN KULUTTUA      000270
39*      TGEN(NRURA)=TGEN(NRURA)+C      000272
40*      GO TO 2489      000275
41*      2411 SAUTO(KAPU)=N      000277
42*      2410 EAUTO(N)=KAPU      000303
43*      IF(EAURA(NRURA),EQ,0) EAURA(NRURA)=N      000314
44*      VAURA(NRURA)=N      000320
45*      SAUTO(N)=0      000322
46*      FLD(16,6,AUOMI(N))=NRURA      000324
47*      2419 DO 2418 I=1,5      000334
48*      2418 PAR(I+1)=LAMBDA(I,NRURA)      000344
49*      PAR(1)=2      000346
50*      PAR(7)=1=LAMBDA(5,NRURA)      000350
51*      TVALI=ABS(EXPJA2(7,PAR))      000353
52*      TGEN(NRURA)=TKELL+TVALI      000363
53*      LKMAUT=LKMAUT+1      000365
54*      C ETSITÄÄN SEURAAVA VAPAANA OLEVA AUTON INDEKSI NRPOIS=TAULUKOSTA      000365
55*      IAPU=LKMAUT+1-(LKMAUT/(NAUTO-1))*(NAUTO-1)      000370
56*      2412 IF (NRPOIS(IAPU),GT,0) GO TO 2415      000400
57*      IAPU=IAPU+1      000407
58*      IF (IAPU,EQ,LKMAUT-(LKMAUT/(NAUTO-1))*(NAUTO-1)) GO TO 2414      000412
59*      IF (IAPU,GE,NAUTO) IAPU=2      000422
60*      GO TO 2412      000431
61*      2414 WRITE(6,2499)      000433
62*      2499 FORMAT('1 KAIKKI AUTOILLE VARATUT INDEKSI OYAT KAYTÖSSÄ )) (FRMT      000437
63*      12499)')/1 TAILUKON NRPOIS DUMPPAUS:1/)      000437
64*      WRITE(6,2498) NRPOIS      000437
65*      2498 FORMAT()      000454
66*      STOP      000454
67*      2415 NRGEN(NRURA)=NRPOIS(IAPU)      000457
68*      NRPOIS(IAPU)=N      000464
69*      C REITIN GENEROINTI KUN SISAANTULOURA ON NRURA      000464
70*      DO 2420 I=1,NREIT      000471
71*      IF (FLD(0,6,URAT(I)),EQ,NRURA) GO TO 2421      000471
72*      2420 CONTINUE      000501
73*      2421 IIRIVI=1      000501
74*      NHAARE=0      000502
75*      DO 2425 I=IRIVI,NREIT      000515
76*      IF (FLD(0,6,URAT(I)),EQ,NRURA) NHAARE=NHAAR+1      000515
77*      2425 CONTINUE      000525
78*      OSUUS=0      000525
79*      SLUKU=RANDOM(0)      000530
80*      DO 2430 I=1,NHAAR      000541
81*      OSUUS=OSUUS+PROS(IRIVI+I-1)      000541
82*      IF (SLUKU,LT,OSUUS) GO TO 2431      000544
83*      2430 CONTINUE      000554
84*      2431 N=NRGEN(NRURA)      000554
85*      FLD(10,6,AUOMI(N))=IRIVI+I-1      000565
86*      C SEURAAVASSA URAHISTEYKSESSÄ KAANTYVA AUTO MERKITAAN      000565
87*      IF (FLD(6,6,URAT(I)),EQ,FLD(6,6,EURA(NRURA))) FLD(0,2,AUOMI(N))=1      000600
88*      FLD(9,1,AUOMI(N))=0      000613

```


09*	C AJONEUVOTYYYPIN GENEROINTI	000613
10*	OSUUS=0	000617
11*	SLUKU=TASAN(0,1,)	000620
12*	DO 2435 I=1,9,4	000631
13*	OSUUS=OSUUS+EPIT(I,NRURA)	000631
14*	2435 IF (SLUKU,LT,OSUUS) GO TO 2436	000634
15*	2436 ITYYP=INT(EPIT(I+3,NRURA)+0,001)	000644
16*	LUOKKA(ITYYP)=LUOKKA(ITYYP)+1	000673
17*	FLD(23,3,AUOMI(N))=ITYYP	000676
18*	C CAR=FOLLOWING=KERTOIMEN GENEROINTI	000676
19*	ALFA(N)=ORJAK(9,1,)	000704
20*	C TULOAIKA MUISTIIN	000704
21*	AU(N,1)=TGEN(NRURA)	000715
22*	C TAVOITENOPFUDEN GENEROINTI	000715
23*	AU(N,2)=500,736.*TASAN(5,1,5)	000720
24*	C HALUTUN JARRUTUSNOPEUDEN GENEROINTI (JARRUVALOJEN NAKEMINEN)	000720
25*	AU(N,3)=ABS(ORJAK(-0,7,0,07))	000726
26*	C MAKSIMIKIIHTYVYYDEN GENEROINTI	000726
27*	AU(N,4)=ABS(ORJAK(1,6,0,1))	000742
28*	C GENEROIDAAN AJONEUVOILLE PITUUDET	000742
29*	AU(N,5)=ABS(ORJAK(EPIT(I+1,NRURA),EPIT(I+2,NRURA)))	000756
30*	C LASKETAAN ASKEN GENEROIDUN AUTON OMINAISUUDET AIKA=ASKELN PAATTYESSA	000756
31*	V(N)=AU(N,2)/2.	001017
32*	DO 2440 I=1,IT	001032
33*	2440 A(N,I)=AU(N,4)/3.	001032
34*	I=0	001034
35*	2450 I=I+1	001036
36*	IF(I=C-TVALI,LT,0) GO TO 2450	001046
37*	T=I-C-TVALI	001056
38*	X(N)= V(N)*T+0,5*A(N,1)*T+T	001064
39*	C KIRJANPITOA	001064
40*	C SEURAAVA SISAAANTULOURA	001105
41*	2489 CONTINUE	001105
42*	RETURN	001105
43*	END	001202

1*	SUBROUTINE LVALOT(LVALO,C,TILA,VAIHE,LV,IVALO,NSKAIS)	000223
2*	INTEGER TILA(2,IVALO),LVALO(IVALO),VAIHE(LV,IVALO),TILA1	000223
3*	LOGICAL TESTI	000223
4*	COMMON/08203/DUMMY(97),TILA1(2,8)	000223
5*	C PÄIVITETÄÄN L-VALOTILATAULUKKO	000223
6*	DO 2510 I=1,IVALO	000223
7*	2510 TILA(2,I)=TILA(2,I)+1	000223
8*	IUUDET=IUUDET+1	000226
9*	2512 TESTI=.FALSE.	000232
10*	IF (IUUDET.LE.0) TESTI=.TRUE.	000232
11*	IAPU=IVALO-NSKAIS	000240
12*	DO 2516 I=1,IAPU	000243
13*	TILA1(1,I)=TILA(1,NSKAIS+I)	000251
14*	2516 TILA1(2,I)=TILA(2,NSKAIS+I)	000252
15*	C JOS EI OLE AIKA VAIHTAA LIIKENNEVALOJA,PALATAAN TAKAISIN	000252
16*	IF (.NOT.TESTI) RETURN	000255
17*	C LIIKENNEVALOJEN UUDELLEEN ASETUS TILA=TAULUKKON	000255
18*	DO 2520 I=1,IVALO	000266
19*	IAPU=TILA(1,I)	000266
20*	IF (IAPU.LE.0) WRITE(6,2590)	000270
21*	2590 FORMAT(' VIRHETILANNE: TAULUKOSSA TILA EI=POSITIIVINEN ALKIO (FRM	000281
22*	IT 2590))	000281
23*	IF (TILA(2,I).LT.VAIHE(IAPU,I)/C) GO TO 2520	000281
24*	TILA(1,I)=IAPU+1	000281
25*	IF (IAPU.GE.LV) TILA(1,I)=1	000284
26*	TILA(2,I)=0	000283
27*	2520 CONTINUE	000285
28*	C LIIKENNEVALOJEN VAIHTO KAISTOITTAIN TAULUKKON LVALO	000285
29*	DO 2549 I=1,IVALO	000285
30*	IAPU=TILA(1,I)	000285
31*	2545 IF (TILA(1,I).LT.5) GO TO 2547	000287
32*	TILA(1,I)=TILA(1,I)-4	000283
33*	GO TO 2545	000286
34*	2547 LVALO(I)=TILA(1,I)	000286
35*	2549 TILA(1,I)=IAPU	000286
36*	C SEURAAVAN UUDELLEENASETUSAJANKOHDAN ETSIMINEN	000286
37*	IUUDET=9999	000286
38*	DO 2550 I=1,IVALO	000287
39*	IAPU=TILA(1,I)	000287
40*	IAPU=VAIHE(IAPU,I)/C=TILA(2,I)	000287
41*	2550 IF (IAPU.LT.IUUDET) IUUDET=IAPU	000287
42*	GO TO 2512	000281
43*	END	000273

1*	SUBROUTINE JONOT(X,V,EAURA,JONO,JONOKA,SAUTO)	000010
2*	REAL X(AUTO),V(AUTO)	000010
3*	INTEGER SAUTO(AUTO),EAURA(NURA),JONO(NSKAIS),JONOKA(NSKAIS)	000010
4*	PARAMETER KPA=1	000010
5*	COMMON/OS201/NREIT,NURA,NOMI ,NSKAIS,NUKAIS,NAUTO	000010
6*	COMMON/OS202/XMIN,TKELL,DUMMY(5)	000010
7*	COMMON/OS204/XSEIS,XLEV,XVALO,DUMMY1(3)	000010
8*	DO 2680 IK=1,NSKAIS	000010
9*	C ETSITÄÄN ENSIMMÄINEN RISTEYKSEEN TULLUT TAI TULEVA AUTO	000010
10*	KAPU=EAURA(IK)	000017
11*	2615 IF (KAPU,LE,3) GO TO 2680	000022
12*	IF (X(KAPU),LT,XSEIS+5,) GO TO 2620	000025
13*	KAPU=SAUTO(KAPU)	000037
14*	GO TO 2615	000041
15*	C LAHINNA RISTEYSTÄ ON AUTO KAPU	000041
16*	2620 IF (V(KAPU),LT,1E=3) GO TO 2630	000043
17*	C JONO ON SAAVUTTANUT MAKSIMIPIITUUTENSA JA MERKITÄÄN TILASTOJA	000043
18*	IF (JONO(IK),EQ,0) GO TO 2625	000051
19*	CALL KIRJA3(JONO(IK),IK)	000053
20*	2625 JONO(IK)=0	000062
21*	GO TO 2680	000062
22*	2630 IF (JONO(IK),LE,0) GO TO 2660	000064
23*	C ETSITÄÄN OLEMASSA OLEVAN JONON LOPPUPAAN AUTO	000064
24*	IF (JONO(IK),LT,2) GO TO 2656	000067
25*	JPITUU =JONO(IK)=1	000074
26*	DO 2655 J=1,JPITUU	000077
27*	IF (SAUTO(KAPU),EQ,0) GO TO 2689	000105
28*	KAPU=SAUTO(KAPU)	000107
29*	2655 IF (V(KAPU),GT,1E=3) GO TO 2689	000113
30*	2656 KAPU=SAUTO(KAPU)	000123
31*	IF (KAPU,EQ,0) GO TO 2680	000126
32*	C JOS LÖYDETYN AUTON NOPEUS ON NOLLA, LIITETÄÄN SE JONOON	000126
33*	2660 IF (V(KAPU),LT,1E=3) JONO(IK)=JONO(IK)+1	000131
34*	GO TO 2680	000140
35*	C KORJATAAN JONON PITUUS JONOSSE TAPAHTUNUTTA MUUTOSTA VASTAAVASTI	000144
36*	C JONO EI KUITENKAAN OLE HAVINNYT KOKONAAN	000144
37*	2689 JONO(IK)=J	000146
38*	2680 CONTINUE	000153
39*	RETURN	000153
40*	END	000206


```

1* SUBROUTINE KIIHTY(X,V,A,ALFA,EAUTO,SAUTO,VAURA,EAURA,XPYS,AU,
2* 1 AUOMI,SURA,LVALO,XURA,XVALI,TNKELT,TNVALI,EURA,
3* 2 KONFLP,KONURA,NKONFL,NKON2,URAT,OHJAUS)
4* COMMON/OS201/AREIT,NURA,NOMI,NSKAIS,NUKAIS,NAUTO
5* COMMON/OS202/XMIN,TKELL,C,LKMAUT, IT,XVAP
6* COMMON/OS204/XSEIS,XLEV,XVALO,AALK,XKAANT,VKAANT
7* REAL X(NAUTO),V(NAUTO),A(NAUTO,IT),ALFA(NAUTO),XPYS(NAUTO),
8* 1AU(NAUTO,NOMI),XURA(NURA),XVALI(NAUTO),TNKELT(20),TNVALI(20)
9* INTEGER EAUTO(NAUTO),SAUTO(NAUTO),EAURA(NURA),VAURA(NURA),
10* 1 EURA(NURA),SURA(NURA),AUOMI(NAUTO),LVALO(NSKAIS),
11* 2 KONFLP(NKONFL),KONURA(NKON2),URAT(NREIT)
12* LOGICAL KONFLI,OHJAUS,KESKEN
13* C MAARITELLAAN ERI KIIHTYVYTYYPPIEN LASKENTAFUNKTIOT
14* DEFINE KIIHT1(N)=AALK*(1-V(N)/AU(N,2))
15* DEFINE KIIHT2(N)=ALFA(N)*(V(KAPU)-V(N))/(XVALI(N))
16* DEFINE KIIHT3(N)=ABS(0.5*V(N)**2/(X(N)-(XPYS(N)+ORJAK(0,0,17))))
17* KESKEN=.FALSE.
18* DO 2701 IK=1,NUKAIS
19* KNFTAL=0
20* I=IK+NSKAIS
21* ISUOR=1
22* C MAETAAN AUTO ULOS VIEVALTA AJOURALTA
23* 2703 IF (EAURA(I),EQ,0) GO TO 2702
24* N=EAURA(I)
25* 2796 IF (OHJAUS) GO TO 2706
26* 2707 IF (KONFLI(X,V,A,AU,AUOMI,XPYS,EAURA,SURA,EURA,XURA,URAT,SAUTO,
27* 1 VAURA,KNFTAL,KONFLP,KONURA,NKONFL,NKON2,TNVALI,N,1)) GO TO 2790
28* GO TO 2782
29* C ONKO KASITELTAVANA OLEVA AJOURA SISAANTULOURA)
30* 2708 IF (I,GT,NSKAIS) GO TO 2727
31* C AUTON PAIKKA SEIS-VIIIVAAN NAHDEN
32* IF (X(N)=XSEIS) 2780,2707,2707
33* C LIIKENNEVALOJEN VARIKOODIT: VIHREA=1, VIH=KELTAINEN=2, PUNAINEN=3,
34* C PUN=KELTAINEN=4
35* 2780 IF (LVALO(I),EQ,2) CALL VIKKEL
36* IF (LVALO(I),EQ,4) CALL PUNKEL
37* IF (LVALO(I),EQ,1,OR,LVALO(I),EQ,3) FLD(2,2,AUOMI(N))=0
38* IF (LVALO(I),EQ,3) GO TO 2789
39* C ONKO KELTAINEN=RUTIIINI ANTANUT LUVAN MENNA)
40* IF (FLD(2,2,AUOMI(N)),EQ,1) GO TO 2789
41* 2782 KAPU=FLD(0,2,AUOMI(N))
42* C AUTO ETENE: 0 = SUORAAN, 1 = VASEMMALLE, 2 = OIKEALLE
43* IF (KAPU,EQ,0) GO TO 2795
44* CALL KAANTY(2795)
45* GO TO 2790
46* 2789 CALL PUNLAH
47* GO TO 2702
48* 2795 KAPU=EAUTO(N)
49* IF (EAUTO(N),EQ,0) KAPU=NAUTO
50* C ONKO AUTO PAKOTETTU TASATSEEN LIIKKEESEEN POISTU=ALIOHJELMASSA)
51* IF (FLD(5,4,AUOMI(N)),NE,6) GO TO 2706
52* A(N,IT)=0
53* GO TO 2790
54* C ONKO AUTO RIIPPUMATON EDELTAJASTAAN?
55* 2706 IF (XVALI(N),LT,XVAP,OR,1.8*ABS((V(KAPU)-V(N))/(
56* 1 (XVALI(N))**2),GT,6E=4) GO TO 2710
57* C VAPAA ETENEMINEN
58* 2705 A(N,IT)=KIIHT1(N)
59* IF (A(N,IT),GT,AU(N,4)) A(N,IT)=AU(N,4)
60* FLD(5,4,AUOMI(N))=1
61* XPYS(N)=0
62* GOTO 2790
63* C CAR=FOLLOWING ETENEMINEN
64* 2710 A(N,IT)=KIIHT2(N)
65* IF (A(N,IT),LT,0) GO TO 2720
66* 2715 IF (KIIHT1(N),LE,A(N,IT)) GO TO 2705
67* 2725 IF (A(N,IT),GE,AU(N,4)) A(N,IT)=AU(N,4)
68* 2755 FLD(5,4,AUOMI(N))=2
69* XPYS(N)=0
70* GOTO 2790
71* 2720 IF (FLD(5,4,AUOMI(KAPU)),EQ,3,OR,A(KAPU,1),LE,-.7) GO TO 2730
72* IF (FLD(5,4,AUOMI(KAPU)),EQ,0,OR,V(KAPU),LE,1E=2) GO TO 2730
73* GO TO 2715
74* 2730 IF (V(N),LE,1E=2) GO TO 2740
75* C ETENEE JAKRUITAEN PYSÄHTYMISPAIKKAANSA ASTI
76* IF (FLD(5,4,AUOMI(N)),EQ,3) GO TO 2750
77* XPYS(N)=XPYS(KAPU)-AU(KAPU,5)
78* IF (XPYS(N),LT,0) XPYS(N)=XURA(I)+XPYS(N)
79* A(N,IT)=KIIHT3(N)
80* IF (A(N,IT),LE,AU(N,3)) GO TO 2760
81* A(N,IT)=0
82* GO TO 2755
83* 2750 A(N,IT)=A(N,IT-1)
84* 2760 FLD(5,4,AUOMI(N))=3
85* GO TO 2790
86* C EI ETENE, ON PYSÄHTYNYT
87* 2740 FLD(5,4,AUOMI(N))=0
88* DO 2741 J=1,IT
89* 2741 A(N,J)=0
90* V(N)=0
91* XPYS(N)=X(N)
92* C KIRJANPITO
93* 2790 IS=FLD(5,4,AUOMI(N))
94* 9 FORMAT(10H AUTOON ,I2,22H SOVELLETAAN YHTALOA ,I2)
95* C ONKO KASITTELEMÄTTÖMIA AUTOJA AJOURALLA?
96* IF (N,EQ,VAURA(I),OR,SAUTO(N),LE,0) GO TO 2702
97* N=SAUTO(N)
98* GO TO 2796

```



```

99* C ONKO AJOURIA KASITTELEMATTA?
100* C ENSIN KASITELLAAN SUORA URA JA SITTEEN SEN RINNALLA OLEVA KAAREVA URA
101* 2702 IF (KESKEN) GO TO 2775
102* IF (ISUOR,LE,NSKAIS) GO TO 2701
103* IKAANT=FLD(6,6,SURA(ISUOR))
104* ISUOR=FLD(6,6,SURA(ISUOR))
105* IF (IKAANT,EQ,0) GO TO 2776
106* IF (FLD(6,6,EURA(IKAANT)),NE,0) GO TO 2776
107* IF (ISUOR,EQ,0) GO TO 2775
108* I=ISUOR
109* KESKEN=TRUE,
110* GO TO 2703
111* 2775 IF (I,EQ,IKAANT) GO TO 2774
112* I=IKAANT
113* KTALL=KNFTAL
114* KNFTAL=0
115* GO TO 2703
116* 2774 IF (FLD(6,6,SURA(IKAANT)),GT,0) I=FLD(6,6,SURA(IKAANT))
117* KESKEN=FALSE,
118* KNFTAL=KTALL
119* IF (I,EQ,IKAANT) GO TO 2702
120* GO TO 2703
121* 2776 I=ISUOR
122* IF (I,EQ,0) GO TO 2701
123* GO TO 2703
124* 2701 CONTINUE
125* RETURN
126* C*****
127* C*****
128* C*****
129* SUBROUTINE VIHKE
130* C ONKO JO AIKAISEMMIN TEHTY PAATOS MENNA Keltaista PAIN)
131* IF (FLD(2,2,AUOMI(N)),EQ,2) RETURN
132* IF (FLD(2,2,AUOMI(N)),EQ,1) RETURN
133* C ONKO AUTO JO AJANUT SEIS-VIIIVAN YLI)
134* IF (X(N)=XSEIS) 3010,3020,3020
135* 3020 IF ( V(N),LT,1E-2) GO TO 3022
136* 3021 XPYS(N)=0
137* FLD(2,2,AUOMI(N))=2
138* RETURN
139* C ONKO AUTO JO AJANUT RISTEYKSEEN)
140* 3022 IF (X(N)=XSEIS=XLEV) 3081,3081,3021
141* C GENEROIDAAN TAVOITEPYSÄHTYMISPAIKKA JA VERRATAAN SITA AUTON SIJAINTIIN
142* 3010 KAPU=EAUTO(N)
143* XPYS(N)=XVALO=XPYS(KAPU)+RANDOM(N)
144* IF (XPYS(N),LT,0) XPYS(N)=XURA(I)+XPYS(N)
145* IF (X(N)=XPYS(N),GT,0) GO TO 3020
146* C LASKETAAN PYSÄHTYMISHIDASTUVUUS JA TESTATAAN SITA
147* A(N,IT)=KIHTI3(N)
148* C ETSITÄÄN TÄULUKOSTA PYSÄHTYMISTODENNAKUISYYS TALLE HIDASTUVUDELLE
149* J=INT(A(N,IT)/0,3048)+1
150* IF (J,GT,20) GO TO 3021
151* IF (RANDOM(0),GT,TNKELT(J)) GO TO 3021
152* 3081 FLD(2,2,AUOMI(N))=1
153* RETURN
154* C*****
155* C*****
156* C*****
157* SUBROUTINE PUNKEL
158* C ONKO JO AIKAISEMMIN PAATTANYT LAHTEA KIIKKEELLE)
159* IF (FLD(2,2,AUOMI(N)),EQ,2) RETURN
160* IF (KONFLI(X,V,A,AU,AUOMI,XPYS,EURA,SURA,EURA,URAT,SAUTO,
161* 1 VAURA,KNFTAL,KONFLP,KONURA,NKONFL,NKON2,INVALI,N,I)) GO TO 3110
162* IF (RANDOM(0),GT,V,5) GO TO 3110
163* C AUTO ON PAATTANYT LAHTEA JO Keltaisen VALON PALAESSA
164* FLD(2,2,AUOMI(N))=2
165* RETURN
166* C AUTO EI LAHDE LIIKKEELLE Keltaisen VALON PALAESSA
167* 3110 FLD(2,2,AUOMI(N))=1
168* RETURN
169* C*****
170* C*****
171* C*****
172* SUBROUTINE PUNLAH
173* C AUTON NYKYINEN LIIKE?
174* IF ( V(N),GT,1E-2 ) GO TO 2910
175* KAPU=EAUTO(N)
176* IF (KAPU,LE,0) KAPU=NAUTO
177* C ONKO AUTO KAUKANA EDELTAJASTAAN JA SEIS-VIIIVASTA?
178* IF (XVALI(N),GT,AU(KAPU,5),AND,XSEIS=X(N),GT,3,) GO TO 2920
179* C AUTO ON PYSÄHTYNYT
180* DO 2905 J=1,IT
181* 2905 A(N,J)=0
182* V(N)=0
183* XPYS(N)=X(N)
184* FLD(5,4,AUOMI(N))=0
185* GO TO 2990
186* C KAUAKSI PYSÄHTYNEEN AUTON KULJETUS LAHEMMAKSI SEIS-VIIIVAA
187* 2920 A(N,IT)=KIHTI1(N)
188* IF (A(N,IT),GT,AU(N,4)/2,) A(N,IT)=AU(N,4)/2,
189* FLD(5,4,AUOMI(N))=1
190* GO TO 2990
191* C ONKO AUTO JO AIKAISEMMIN ALKANUT PYSÄHTYÄ?
192* 2910 IF (FLD(5,4,AUOMI(N)),NE,3) GO TO 2930
193* A(N,IT)=A(N,IT-1)
194* GO TO 2990

```

```

001022
001022
001024
001025
001032
001036
001040
001042
001046
001050
001052
001054
001056
001061
001063
001065
001066
001070
001101
001102
001104
001110
001112
001113
001115
001120
001120
001120
001120
001213
001213
001230
001234
001242
001250
001257
001261
001271
001271
001275
001275
001305
001310
001323
001356
001356
001365
001403
001414
001420
001432
001441
001441
001441
001441
001477
001477
001477
001514
001514
001540
001544
001553
001563
001563
001567
001576
001576
001576
001576
001652
001652
001652
001661
001665
001665
001720
001720
001753
001753
001754
001756
001761
001765
001765
001767
002005
002026
002036
002036
002040
002047
002063

```



```

195* C TUTKITAAN, ALKAAKO AUTO PYSÄHTYÄ TALLA AIKA=ASKELEELLA 002063
196* 2930 IF (ABS(XPYS(N)-XSEIS),GT,2, ) XPYS(N)=XVALO=ORJAK(0,,0,1) 002065
197* A(N,IT)=KIIHT3(N) 002105
198* IF (A(N,IT),LE,AU(N,3) ) GO TO 2940 002136
199* IMTCH=1 002146
200* GO TO 2920 002150
201* C AUTO ALKAA PYSÄHTYÄ 002150
202* 2940 FLD(5,4,AUOMI(N))=3 002152
203* GO TO 2990 002161
204* C HÄYÄN SEURAAVA AUTO AJOURALTA KASITELTAVAKSI 002161
205* 2951 KAPU=N 002163
206* N=SAUTO(N) 002164
207* IF (N,LE,0) RETURN 002170
208* C ONKO AUTO EDELTAJASTAAN RIIPPUVA? 002170
209* IF (XVALI(N),LT,XVAP) GO TO 2960 002177
210* A(N,IT)=KIIHT1(N) 002220
211* C AUTO KULKEE VAPAAN ETENEMISEN TAI MAKSIMI KIIHTYVYYDEN MUKAAN 002220
212* 2950 IF (A(N,IT),GT,AU(N,4)/2, ) A(N,IT)=AU(N,4)/2, 002226
213* FLD(5,4,AUOMI(N))=1 002253
214* GO TO 2990 002263
215* C AUTO KULKEE EDELTAANSA SEURATEN 002263
216* 2960 IF (KIIHT2(N),LE,0) GO TO 2961 002265
217* 2962 A(N,IT)=KIIHT1(N) 002304
218* C AUTO KULKEE CAR-FOLLOWINGIN TAI TAVOITENOPEUDEN ANTAMAN KIIHTYVYYDEN MUKAAN 002304
219* IF (KIIHT2(N),GE,A(N,IT)) GO TO 2950 002322
220* A(N,IT)=KIIHT2(N) 002340
221* C JOS CAR-FOLLOWING KIIHTYVYYS ALITTAÄ TAI YLITTAÄ AUTON MINIMI= TAI MAKSIMI= 002340
222* C KIIHTYVYYDET, NOUDATETAAN VASTAÄVÄÄ RAJAKIIHTYVYYTTÄ 002340
223* IF (A(N,IT),LE,2*AU(N,3)) A(N,IT)=2*AU(N,3) 002360
224* IF (A(N,IT),GE,AU(N,4)/2, ) A(N,IT)=AU(N,4)/2, 002406
225* 2963 FLD(5,4,AUOMI(N))=2 002422
226* GO TO 2990 002431
227* C ONKO EDELTAJA PYSÄHTYNYT TAI PYSÄHTYMASSA? 002431
228* 2961 IF (FLD(5,4,AUOMI(KAPU)),EQ,3,OR,FLD(5,4,AUOMI(KAPU)),EQ,0) 002433
229* 1 GO TO 2970 002433
230* GO TO 2962 002451
231* C ONKO KASITELTÄVÄÄ OLEVA AUTO PYSÄHTYNYT TAI PYSÄHTYMASSA? 002451
232* 2970 IF (V(N),GT,1E-2) GO TO 2971 002453
233* XPYS(N)=X(N) 002461
234* FLD(5,4,AUOMI(N))=0 002467
235* DO 2975 J=1,IT 002510
236* 2975 A(N,J)=0 002510
237* GO TO 2990 002511
238* 2971 IF (FLD(5,4,AUOMI(N)),NE,3) GO TO 2972 002513
239* A(N,IT)=A(N,IT-1) 002522
240* GO TO 2990 002536
241* C LASKETAÄ PYSÄHTYMISPAIKKA JA PYSÄHTYMISEEN TARVITTAVA JARRUTUSHIDASTUVUUS 002536
242* 2972 XPYS(N)=XPYS(KAPU)=AU(KAPU,5) 002540
243* IF (XPYS(N),LT,0) XPYS(N)=XURA(I)+XPYS(N) 002552
244* A(N,IT)=KIIHT3(N) 002567
245* IF (A(N,IT),LE,AU(N,3) ) GO TO 2973 002623
246* XPYS(N)=999999 002627
247* A(N,IT)=0 002631
248* GO TO 2963 002632
249* 2973 FLD(5,4,AUOMI(N))=3 002634
250* C KIRJANPITOÄ 002634
251* 2990 IS=FLD(5,4,AUOMI(N)) 002644
252* 2999 FORMAT(10H AUTOON ,12,22H SOVELLETAAN YHTÄLÖÄ ,12,1+1) 002651
253* GO TO 2951 002651
254* C***** 002651
255* C***** 002651
256* C***** 002651
257* SUBROUTINE KAANTY(R) 002745
258* C AUTON OLLESSA KAUKANA KAANTYMISKOHDASTA NOUDATETAAN YL KIIHTYVYYDEN RUTIIINIA 002745
259* NRREIT=FLD(10,6,AUOMI(N)) 002745
260* IF (FLD(6,6,EURA(I)),EQ,0) RETURN 002753
261* C TUTKITAAN ONKO KO URAAN EDESSÄ OLEVA KAANTYVA URA AUTON REITILLÄ 002775
262* DO 3210 J=0,5 002775
263* 3210 IF (FLD(6,6,URAT(NRREIT)),EQ,FLD(6,6,EURA(I))) GO TO 3220 002775
264* RETURN 003007
265* 3220 IF (FLD(5,4,AUOMI(N)),EQ,4) GO TO 3260 003014
266* C ONKO AUTO ALKANUT JO HIDASTAA TULEVAN KAANTYMISEN VUOKSI 003014
267* IF (FLD(5,4,AUOMI(N)),EQ,5) GO TO 3250 003023
268* C ONKO AUTO OHITTANUT KAANTYMISEN ALKUKOH DAN) 003023
269* IF (XURA(I)=X(N),LT,XKAANT) GO TO 3260 003050
270* A(N,IT)=0,5*(VKAANT+2=V(N)+2)/(XURA(I)=X(N)) 003056
271* IF (A(N,IT),GT,AU(N,3)) RETURN 003273
272* C KUN KAANTYMISJARRUTUSHIDASTUVUUS ON PIENEMPI KUIN HALUTTU JARRUTUSHIDASTUVUUS, 003273
273* C NOUDATETAAN KAANTYMISJARRUTUSHIDASTUVUUTTA 003073
274* FLD(5,4,AUOMI(N))=5 003106
275* GO TO 3270 003115
276* C JOS AUTO ON OHITTANUT KAANTYMISEN ALKUKOH DAN, NOUDATETAAN ENTISIA ARVOJA 003115
277* 3250 IF (XURA(I)=X(N),LT,XKAANT) GO TO 3260 003117
278* A(N,IT)=A(N,IT-1) 003130
279* GO TO 3270 003144
280* C AUTO ON JUURI KAANTYMASSA JA LASKETAAN KAANTYMISKIIHTYVYYS 003144
281* 3260 IF (V(N)+AU(N,4)*C,GT,VKAANT) GO TO 3261 003106
282* A(N,IT)=AU(N,4) 003162
283* GO TO 3262 003171
284* 3261 A(N,IT)=(VKAANT=V(N))/C 003173
285* 3262 FLD(5,4,AUOMI(N))=4 003206
286* C TARKISTETAAN, AIHEUTTAÄKO EDESSÄ OLEVA AUTO PIENEMMAN KIIHTYVYYSARVON 003206
287* 3270 IF (XVALI(N),GT,XVAP) RETURN 003216
288* IF (A(N,IT),LT,KIIHT2(N)) RETURN 003231
289* A(N,IT)=KIIHT2(N) 003270
290* FLD(5,4,AUOMI(N))=2 003311
291* RETURN 003321
292* END 003364

```



```

1* LOGICAL FUNCTION KONFLI(X,V,A,AU,AUOMI,XPYS,EURA,SURA,EURA,XURA, 000051
2* 1 URAT,SAUTO,VAURA,KNF TAL,KONFLP,KONURA,NKONFL,NKON2,TNVALI,N,MT) 000051
3* PARAMETER NTUTAU=1,XLEV=3 000051
4* COMMON/OS201/NREIT,NURA,NOMI,NSKAI5,NUKAI5,NAUTO 000051
5* COMMON/OS202/XMIN,TKELL,C,DUMMY,IT,XVAP 000051
6* COMMON/OS206/SEKTOR(8) 000051
7* REAL X(NAUTO),V(NAUTO),A(NAUTO,IT), XPYS(NAUTO), 000051
8* 1AU(NAUTO,NOMI),XURA(NURA), TNVALI(20) 000051
9* INTEGER EAUURA(NURA),SURA(NURA),AUOMI(NAUTO),SAUTO(NAUTO), 000051
10* 1 VAURA(NURA),EURA(NURA),KONURA(NKON2),KONFLP(NKONFL),URAT(NREIT) 000051
11* INTEGER HAVAIN(100) 000051
12* LOGICAL HYVVAL 000051
13* DEFINE VARV1(VTOD)=2,+.3,4*A LOG(VTOD+1)+0,61*VTOD+ 000051
14* 1 ORJAK(0,.,26*VTOD) 000051
15* DEFINE VARV(VTOD)=AMAX1(0,0001,VARV1(VTOD)) 000051
16* DEFINE XARV1(XTOD)=10,*(0,56+,12*EXP(XTOD/10,)+1E+7+1,3*XTOD/10,)+ 000051
17* 1 ORJAK(0,.,2*XTOD) 000051
18* DEFINE XARV(XTOD)=AMAX1(.,0001,XARV1(XTOD)) 000051
19* DEFINE KIHTI3(N)=ABS(0,5*V(N)+2/(X(N)-(XPYS(N)+ORJAK(0,.,0,2)))) 000051
20* IF (NTUTKO,EQ,0) READ(5,2899) NTUTKO 000051
21* 2899 FORMAT(I2) 000061
22* M=MT 000061
23* KONFLI=.FALSE, 000063
24* IRIST=0 000064
25* XKLISA=0, 000065
26* KONLUK=1 000066
27* C TARKASTELTAVANA ON AJONEUVO N AJOURALLA MT 000066
28* C HAETAAN AUTON EDESSA OLEVA KONFLIKTIPISTE NRPIST 000066
29* J=1 000070
30* MTALL=M 000072
31* 2801 JUTLUK=1 000074
32* 2805 J=J+1 000076
33* IF (J,GT,4) GO TO 2804 000100
34* NRPIST=FLD(7*J,7,KONURA(M)) 000104
35* IF (NRPIST,GT,NKONFL) GO TO 2803 000114
36* IF (NRPIST,EQ,0) GO TO 2804 000120
37* K=1 000122
38* 2802 K=K+1 000125
39* IF (K,GE,2) WRITE(6,2892) 000127
40* 2892 FORMAT(' VIRHETILANNE: RISTIRIITA TULUKOIDEN KONURA TAI KONFLP Y 000141
41* 1IEDOISSA (FRMT 2892)') 000141
42* IF (K,GE,2) STOP 000141
43* IF (MTALL,NE,FLD(6*K,6,KONFLP(NRPIST))) GO TO 2802 000156
44* XKONFL=FLD(12*7*K,7,KONFLP(NRPIST)) 000167
45* IF (X(N)=XKONFL+XKLISA) 2806,2806,2805 000200
46* C ETSITAAN KONFLIKTIPISTE URAAN M JATKOKETJUSTA 000200
47* 2803 MTALL=M 000210
48* M=NRPIST+NKONFL+NURA 000211
49* J=1 000215
50* GO TO 2805 000217
51* C ETSITAAN KONFLIKTIPISTE URAAN M EDELTAVASTA URASTA, KOSKA URASTA M EI LOYTNYNYT 000221
52* 2804 M=MTALL 000222
53* XKLISA=XKLISA+XURA(M) 000227
54* M=FLD(0,6,EURA(M)) 000227
55* C JOS EI LOYDY AUTON EDESTA KONFLIKTIPISTETTA, EI SYNNY KONFLIKTINUMKAA 000227
56* IF (M,EQ,0) GO TO 2890 000233
57* MTALL=M 000235
58* J=1 000237
59* GO TO 2801 000241
60* C JOS AUTON EDESTA LOYTNYNYTTA K-PISTETTA LAHESTYY SAMASTA SUUNNASTA JOKIN 000241
61* C TOINEN KONFLIKTISSA OLEVA AUTO, NOUDATETAAN YLEISTA KIIHTYVYYSRUTIIINIA TALLE 000241
62* C AUTOLLE 000241
63* 2806 IF (KNFTAL,EQ,NRPIST) RETURN 000243
64* C AUTON EDESSA ON KONFLIKTIPISTE, RISTEAVA AJOURA ON NIMELTAAN IRIST 000251
65* IRIST=FLD(6*(1-K),6,KONFLP(NRPIST)) 000262
66* XTOD1=XTOD(N,NRPIST) 000267
67* IF (IRIST,LE,NURA) GO TO 2820 000274
68* CALL JALKNF 000276
69* IF (KONFLI) RETURN 000303
70* GO TO 2890 000303
71* C JOS RISTEAVA URA PAATTYY K-PISTEESEEN, ETSITAAN KONFLIKOIVAA AUTOA ENSIN SITA 000303
72* C EDELTAVASTA URASTA 000303
73* 2820 I=1 000305
74* IAPU=IRIST 000306
75* 2822 I=I+1 000311
76* IF (I,GT,4) WRITE(6,2893) IRIST,NRPIST 000313
77* 2893 FORMAT(' VIRHETILANNE: RISTEAVAN URAAN, I4, I URA LUETTELOSTA EI LOY 000332
78* 1TYNYTKAAN KONFLIKTIPISTETTA',I4, I (FRMT 2893)') 000332
79* IF (I,GT,4) STOP 000332
80* IF (FLD(1*7,7,KONURA(IAPU)),EQ,NRPIST) GO TO 2825 000341
81* IF (FLD(1*7,7,KONURA(IAPU)),LE,NKONFL) GO TO 2822 000352
82* IAPU=FLD(1*7,7,KONURA(IAPU))+NKONFL+NURA 000363
83* I=1 000373
84* GO TO 2822 000375
85* 2825 IF (I,EQ,4) GO TO 2826 000377
86* IF (FLD((I+1)*7,7,KONURA(IAPU)),NE,0) GO TO 2807 000402
87* C NYT KASITELLAAN ENSIN VIIMEINEN AUTO EURALTA 000402
88* 2826 KAPU=FLD(0,6,EURA(IRIST)) 000415
89* IF (KAPU,EQ,0) GO TO 2821 000420
90* IF (VAURA(KAPU),EQ,0) GO TO 2821 000422

```



```

91*      KAPU=VAURA(KAPU)
92*      IF (X(KAPU),LT,AU(KAPU,5)+XLEV/2,) GO TO 2828
93*      2821 KAPU=FLD(6,6,EURA(IRIST))
94*      IF (KAPU,EQ,FLD(8,6,EURA(MTALL)),OR,KAPU,EQ,FLD(6,6,EURA(MTALL)))
95*      1      GO TO 2807
96*      IF (KAPU,EQ,0) GO TO 2807
97*      IF (VAURA(KAPU),EQ,0) GO TO 2807
98*      KAPU=VAURA(KAPU)
99*      IF (X(KAPU),GT,AU(KAPU,5)+XLEV/2,) GO TO 2807
100*      2828 XTOD2=0
101*      GO TO 2815
102*      C ETSITAAN RISTEAVALTA AJOURALTA ENSIMMAINEN AUTO, JOS EI LOYDY, TUTKITAAN VIELA
103*      C SEURAAVAKIN AJOURA
104*      2807 KAPU=EAURA(IRIST)
105*      IF (KAPU,NE,0) GO TO 2809
106*      IRIST=FLD(8,6,SURA(IRIST))
107*      C RISTEAVALTA AJOURALTA EI LOYDY AJONEUVOA
108*      IF (IRIST,EQ,0) GO TO 2890
109*      GO TO 2807
110*      C TUTKITAAN RISTEAVAN AUTON PAIKKA KONFLIKTIPISTEEN SUMTEEN
111*      2809 XTOD2=XTOD(KAPU,NRPIST)
112*      IF (XTOD2+AU(KAPU,5)+XLEV/2.,GT,0) GO TO 2815
113*      C OTETAAN SEURAAVA AUTO, KOSKA ASKEN TUTKITTU AUTO ON JO AJANUT K=PISTEEN OHI
114*      2810 KAPU=SAUTO(KAPU)
115*      IF (KAPU,EQ,0) GO TO 2890
116*      GO TO 2809
117*      C LASKETAAN ARVIOIDUT AJOAJAT MOLEMMILLE AUTOILLE
118*      C AUTO1 ON KASITELTAVANA OLEVA AUTO JA AUTO2 ON RISTEAVAA AJOURAA PITKIN
119*      C ETENEVA AUTO
120*      2815 TARV1=XARV(XTOD1)/VARV(V(N))
121*      TARV2=XARV(XTOD2)/VARV(V(KAPU))
122*      IF (XTOD1,LE,0) TARV1=0,
123*      IF (XTOD2,LE,0) TARV2=0,
124*      C JOS RISTEAVAN AUTON ARVIOITU AJOAIKA >> KO AUTON AJOAIKA, EI OLE KONFLIKTIN=
125*      C UHKA
126*      IF (TARV1,LT,TARV2/5,) GO TO 2890
127*      C RISTEAVA AUTO LAHESTYY KASITELTAVAN AUTON NAKEMASEKTORISSA ISEKTO
128*      ISEKTO=FLD(26+3*(1-K),3,KONFLP(NRPIST))
129*      C ONKO AUTO HAVAINNUT RISTEAVAA URAA PITKIN ETENEVAN AUTON)
130*      IF (FLD((JUTLUK-1)*18,18,HAVAIN(N)),EQ,KAPU) GO TO 2830
131*      ALFA=3,14159*SEKTOR(ISEKTO)/180,
132*      HAVKUL=ASIN(XTOD2*SIN(ALFA))/SORT(XTOD1*XTOD1+XTOD2*XTOD2-XTOD1*
133*      1XTOD2+COS(ALFA))
134*      TNHAV=COS(HAVKUL/2,)*EXP(-XTOD1/30,)*EXP(-XTOD2/20,)
135*      IF (XTOD1,GT,FLD(32,4,KONFLP(NRPIST)),AND,XTOD2,GT,FLD(32,4,KONFLP
136*      1(NRPIST))) TNHAV=0
137*      IF (RANDOM(0),GT,TNHAV) GO TO 2880
138*      FLD((JUTLUK-1)*18,18,HAVAIN(N))=KAPU
139*      C ETUAJO-OIKEUDEN MAARAAMINEN LAHESTYMISSSEKTORIN AVULLA
140*      2830 IF (ISEKTO,GE,4) GO TO 2840
141*      C AIKAVALIN HYVAKSYMISEN TESTAUS, KUN AUTO EI OLE ETUAJO-OIKEUTETTU
142*      IF (HYVVAL(TARV1,TARV2)) GO TO 2880
143*      XPYS(N)=FLD(12+7*K,7,KONFLP(NRPIST))-XLEV/2,
144*      GO TO 2850
145*      C ETUAJO-OIKEUTETUN AUTON AIKAVALIN HYVAKSYMISEN TESTAUS
146*      2840 IF (HYVVAL(TARV1,TARV2)) GO TO 2880
147*      XPYS(N)=FLD(12+7*K,7,KONFLP(NRPIST))-XLEV/2,
148*      IF (KIINT3(N),GT,2*AU(N,3)) GO TO 2880
149*      C UHKAAVAN KONFLIKTIN VUOKSI ANNETAAN AUTOLLE JARRUTUSHIDASTUVUUSARVO
150*      2850 A(N,IT)=KIINT3(N)
151*      FLD(22,1,AUOMI(N))=1
152*      FLD(5,4,AUOMI(N))=3
153*      KONFLI=.TRUE,
154*      KNFTAL=NRPIST
155*      C JOS AUTO ON PYSÄHTYNYT KAUAKSI K=PISTEESTA JA EDELTÄJÄSTÄÄN, SE AJAA ETEENPÄIN
156*      IF (V(N),GT,1E-2) GO TO 2870
157*      IF (XPYS(N)=X(N),LT,3) GO TO 2870
158*      A(N,IT)=2*(1-V(N)/AU(N,2))
159*      IF (A(N,IT),GT,AU(N,4)/2,) A(N,IT)=AU(N,4)/2,
160*      FLD(5,4,AUOMI(N))=1
161*      RETURN
162*      2870 CALL KIRJ2(N,X,V,A(N,IT),A(KAPU,IT),KNFTAL,NAUTO,TKELL,KAPU,
163*      1      XTOD1,XTOD2,AUOMI)
164*      RETURN
165*      C TUTKITAAN KONFLIKTIMÄHDOLLISUUTTA SEURAAVANKIN AUTON KANSSA
166*      2880 KAPU=SAUTO(KAPU)
167*      IF (KAPU,LE,0) GO TO 2890
168*      JUTLUK=JUTLUK+1
169*      IF (JUTLUK,LE,NTUTAU) GO TO 2880
170*      C TUTKITAAN SEURAAVA KONFLIKTIPISTE, JOKA ON NRPISTIEEN EDESSA
171*      2890 KONLUK=KONLUK+1
172*      FLD(22,1,AUOMI(N))=0
173*      IF (KONLUK,GT,NTUTKO) RETURN
174*      IF (M,EQ,0) RETURN
175*      GO TO 2801
176*      C*****
177*      C*****
178*      C*****

```



```

179*      FUNCTION HYVVAL(TARV1,TARV2)
180*      IF (TARV1.GT.3) GO TO 3310
181*      C KASITELTAVAN AUTON AJOAika ON PIENI
182*      L=INT(2*TARV2)+1
183*      IF (L.GT.20) GO TO 3305
184*      HYVVAL=FALSE
185*      IF (RANDOM(0).GT.TNVALI(L)) RETURN
186*      3305 HYVVAL=TRUE
187*      RETURN
188*      C KASITELTAVA AUTO EI OLE AIVAN LAHELLA KONFLIKTIPIISTETTA
189*      3310 HYVVAL=FALSE
190*      IF (TARV1.LT.TARV2) HYVVAL=TRUE
191*      RETURN
192*      C*****
193*      C*****
194*      C*****
195*      FUNCTION XTOD(N,KPIST)
196*      C AUTON NYKYINEN AJOURA OLKOON IURA
197*      IURA=FLD(16,6,AUOMI(N))
198*      C ONKO KONFLIKTIPIISTE SAMALLA URALLA KUIN AUTO)
199*      XTOD=XURA(IURA)=X(N)
200*      JURA=IURA
201*      3442 DO 3440 I=0,4
202*      3440 IF (KPIST.EQ.FLD(7*I,7,KONURA(JURA)))GO TO 3470
203*      IF (FLD(20,7,KONURA(JURA)).LE.NKONFL) GO TO 3460
204*      JURA=FLD(20,7,KONURA(JURA))
205*      GO TO 3442
206*      C OTETAAN AUTON REITILTA SEURAAVA URA
207*      3460 IREIT=FLD(10,6,AUOMI(N))
208*      DO 3461 IR=0,5
209*      IF (IURA.EQ.FLD(6*IR,6,URAT(IREIT))) GO TO 3462
210*      CONTINUE
211*      WRITE(6,3490)
212*      3490 FORMAT(' VIRHETILANNE: AUTO ON POISSA REITILTÄÄN (FRMT 3490)')
213*      STOP
214*      3462 IR=IR+1
215*      IF (IR.GT.5) GO TO 3463
216*      IURA=FLD(6*IR,6,URAT(IREIT))
217*      IF (IURA.GT.0) GO TO 3442
218*      C AUTON REITILTA EI LOYTUNYT KO KONFLIKTIPIISTETTÄ, JOTEN AUTO KÄNTYY ENNEN SITÄ
219*      3463 XTOD=99
220*      RETURN
221*      3470 L=L+1
222*      3471 L=L+1
223*      IF (IURA.NE.FLD(6*L,6,KONFLP(KPIST))) GO TO 3471
224*      XTOD=XTOD+(XURA(IURA)=FLD(12+7*L,7,KONFLP(KPIST)))
225*      IF (L.LT.2) RETURN
226*      WRITE(6,3492)
227*      3492 FORMAT(' VIRHETILANNE: RISTIRIITA TAULUKOSSA KONFLP JA URANUMERON
228*      1 VALILLA (FRMT 3492)')
229*      STOP
230*      C*****
231*      C*****
232*      C*****
233*      SUBROUTINE JALKNF
234*      INTEGER TILA
235*      LOGICAL OHJAUS
236*      COMMON/OS203/OHJAUS,SKALA(8,2),TNJAL(4,20),TILA(2,6)
237*      JKNFU=IRIST=NURA
238*      C TUTKITAAN, PALAAKO L-VALO SUUNTAAN JKNFU JA KAUANKO SE ON PALANUT
239*      IF (.NOT.OHJAUS) GO TO 3530
240*      I=IFIX(TILA(1,JKNFU)/2,+0,5001)
241*      3510 IF (I.LE.2) GO TO 3515
242*      I=I-2
243*      GO TO 3510
244*      3515 IF (I.GT.0) GO TO 3520
245*      WRITE(6,3590) I
246*      3590 FORMAT(' VIRHETILANNE: JALANKULKIJAN LIIKENNEVALOKOODI EI VOI OLL
247*      1A =1,10,1 (FRMT 3590)')
248*      STOP
249*      3520 IND=INT(TILA(2,JKNFU)+C/SKALA(JKNFU,I)+1+(I-1)*10)
250*      IF (IND.LT.1) IND=1
251*      IF (IND.GT.20) IND=20
252*      C LASKETAAN, MONESKO KONFLIKTIPIISTE ON KASITELTAVANA KADUN REUNASTA LUKIEN
253*      I=FLD(19,7,KONFLP(NRPIST))
254*      C ARVOTAAN, ONKO KONFLIKTIPIISTEISSA JALANKULKIJA
255*      3530 IF (FLD(0,10,HAVAIN(N)).EQ.999) GO TO 3540
256*      IF (OHJAUS) GO TO 3535
257*      IF (RANDOM(0).GT.0.5) RETURN
258*      GO TO 3540
259*      3535 IF (RANDOM(0).GT.TNJAL(I,IND)) RETURN
260*      C AUTO PYSÄHTYY KONFLIKTIPIISTEISSÄ OLEVAN JALANKULKIJAN VUOKSI, JOS TARVITTAVA
261*      C JARRUTUSHIDASTUVUUS ON RAJAJARRUTUSTA PIENEMPI
262*      3540 FLD(0,10,HAVAIN(N))=999
263*      XPYS(N)=FLD(12,7,KONFLP(NRPIST))
264*      IF (KIIHT3(N).GT.AU(N,3)) RETURN
265*      A(N,1)=KIIHT3(N)
266*      FLD(22,1,AUOMI(N))=1
267*      FLD(5,4,AUOMI(N))=3
268*      KONFLI=.TRUE
269*      KNFTALE=NRPIST
270*      CALL KIRJ2(N,X,V,A(N,1),0,KNFTAL,NAUTO,TKELL,999,XTOD1,0,AUOMI)
271*      RETURN
272*      END

```



```

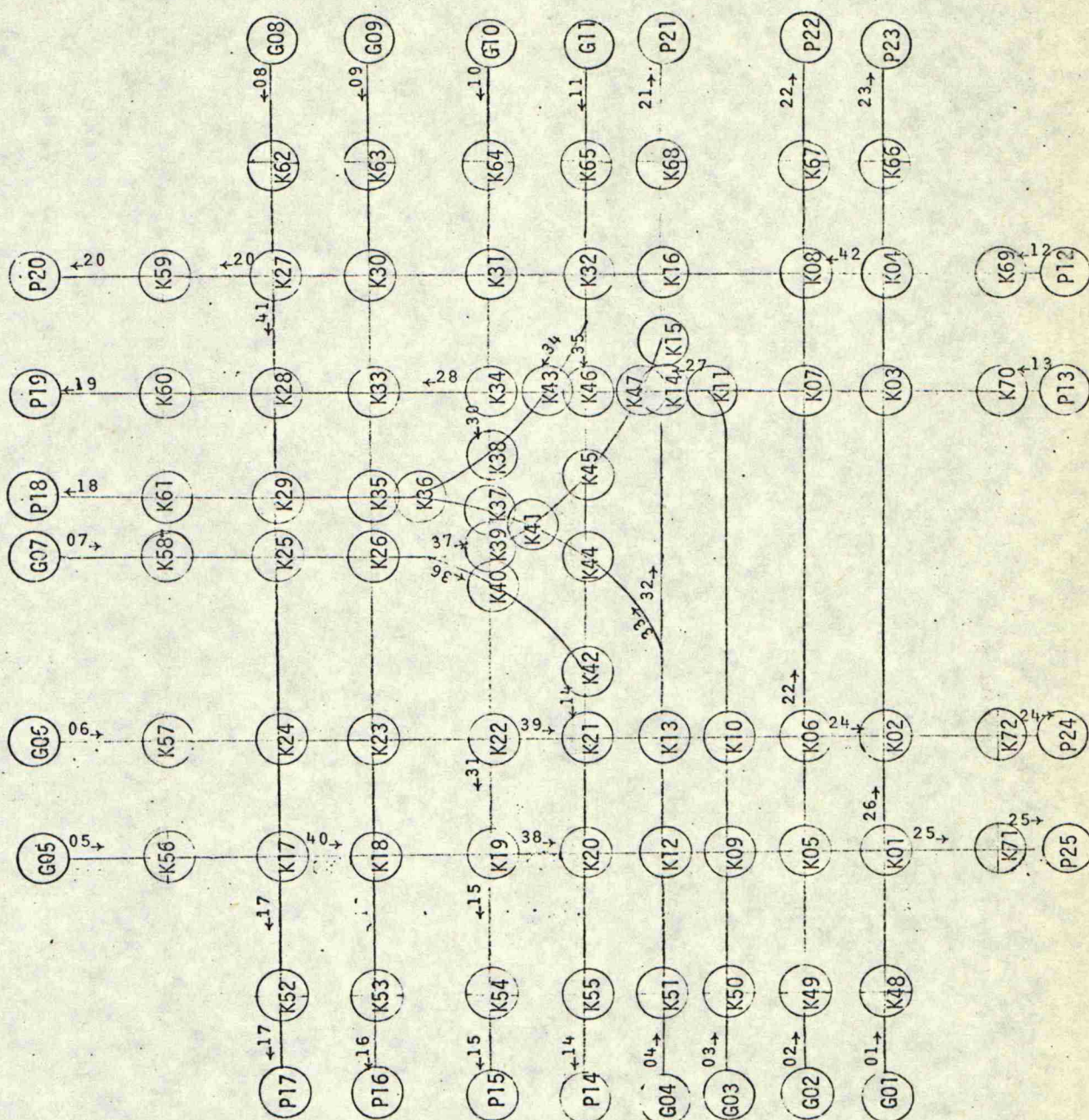
1*      SUBROUTINE KIRJA
2*      REAL KNFKEL(100),KNFVAL(100),KNFNOP(100),TALLE(100,7)
3*      INTEGER KNFLKM(100),KALLE(100,4),KNFTOT(100),JONO(25),
4*      1      JONLKM(25),KNFAPU(25),KNFHID(10,100),KNFKES(6,6,10,2)
5*      COMMON/OS207/KIRITUL(10),MIELKF(9),MRAJA,A1PIEN,A1SUUR
6*      COMMON/OS208/KNFPIS(2,100)
7*      COMMON/OS209/NKONFL
8*      C *****
9*      C *****
10*     C *****
11*     ENTRY KIRJAZ(N,X,V,A1,A2,KNFTAL,NAUTO,TKELL,KAPU,XTOD1,XTOD2,
12*     1      AUOMI)
13*     REAL X(NAUTO),V(NAUTO)
14*     INTEGER AUOMI(NAUTO)
15*     V1=V(N)
16*     V2=0
17*     IF (KAPU,LE,NAUTO) V2=V(KAPU)
18*     LK1=FLD(23,3,AUOMI(N))
19*     LK2=6
20*     IF (KAPU,LE,NAUTO) LK2=FLD(23,3,AUOMI(KAPU))
21*     C TESTATAAN, ONKO TAMA KONFLIKTI OTETTU TILASTOIHIN JO AIKAISEMMIN
22*     IF (KALLE(KNFTAL,1),EQ,N,AND,KALLE(KNFTAL,2),EQ,KAPU) GO TO 4010
23*     IF (KALLE(KNFTAL,1),EQ,0) GO TO 4025
24*     ISW=2
25*     4000 IF (TALLE(KNFTAL,5),GT,-0.7) GO TO 4004
26*     CALL TILAST(KALLE(KNFTAL,1),KALLE(KNFTAL,2),KALLE(KNFTAL,3),
27*     1      KALLE(KNFTAL,4),KNFTAL,TALLE(KNFTAL,1),TALLE(KNFTAL,2)
28*     2      ,TALLE(KNFTAL,3),TALLE(KNFTAL,4),TALLE(KNFTAL,5),
29*     3      TALLE(KNFTAL,6),TALLE(KNFTAL,7))
30*     4004 IF (ISW,EQ,2) GO TO 4005
31*     KALLE(KNFTAL,1)=0
32*     RETURN
33*     4005 KALLE(KNFTAL,1)=N
34*     KALLE(KNFTAL,2)=KAPU
35*     KALLE(KNFTAL,3)=LK1
36*     KALLE(KNFTAL,4)=LK2
37*     TALLE(KNFTAL,1)=XTOD1
38*     TALLE(KNFTAL,2)=XTOD2
39*     TALLE(KNFTAL,3)=V1
40*     TALLE(KNFTAL,4)=V2
41*     TALLE(KNFTAL,5)=A1
42*     TALLE(KNFTAL,6)=A2
43*     TALLE(KNFTAL,7)=TKELL
44*     RETURN
45*     C TESTATAAN, ONKO AIEMMIN TILASTOITU KONFLIKTI NYT VAARALLISEMPI
46*     4010 IF (A1,LT,TALLE(KNFTAL,5)) GO TO 4005
47*     ISW=1
48*     GO TO 4000
49*     C *****
50*     C *****
51*     C *****
52*     ENTRY KIRJAZ(JONMAX,IK)
53*     JONO(1K)=JONO(1K)+JONMAX
54*     JONLKM(1K)=JONLKM(1K)+1
55*     RETURN
56*     C *****
57*     C *****
58*     C *****
59*     ENTRY KIRJAZ(R,TKELL)
60*     WRITE(6,4094) TKELL
61*     4094 FORMAT('1 TULOSTETAAN KUMULATIIVISIA TILASTOJA, KUN SIMULOINTI ON
62*     1 KESTANYT',F6.0,'1 SEKUNTIA'//
63*     2      '1 KONFLIKTI= KONFLIKTIEN KESKIMAAR, KESKIMAAR, JAR
64*     3RUTUSHIDASTUVUUSJAKAUTUMAT//
65*     41 PISTE LUKUMAARA TAPAHTUMIS= NOPEUS ENNEN <=9,0 <=
66*     50,0 <=7,0 <=6,0 <=5,0 <=4,0 <=3,0 <=2,0 <=1,0 <=0,0//
67*     630X,'AIKAVALI KONFLIKTIA'//)
68*     DO 4050 I=1,NKONFL
69*     EKNFV=KNFVAL(I)/KNFTOT(I)
70*     EKNFN=KNFNOP(I)/KNFTOT(I)
71*     4050 WRITE(6,4092) I,KNFTOT(I),EKNFV,EKNFN,(KNFHID(J,I),J=1,10)
72*     4092 FORMAT(19,113,F13.1,F14.1,4X,10I7)
73*     NKNF=0
74*     DO 4051 I=1,NKONFL
75*     4051 NKNF=NKNF+KNFTOT(I)
76*     DO 4053 J=1,10
77*     KNFAPU(J)=0
78*     DO 4052 I=1,NKONFL
79*     4052 KNFAPU(J)=KNFAPU(J)+KNFHID(J,I)
80*     4053 CONTINUE
81*     WRITE(6,4095) NKNF,(KNFAPU(J),J=1,10)
82*     4095 FORMAT(3X,122('1')/3X,'YHTEENSA',111,31X,10I7)
83*     KERRAT=0
84*     JRAJA=2
85*     IF (A1SUUR,GE,0) JRAJA=1
86*     DO 4061 I=1,MRAJA
87*     DO 4060 J=1,JRAJA
88*     AR=IABS(J-2)*A1PIEN+(J-1)*A1SUUR
89*     IF ((KERRAT/2)+2,EQ,KERRAT) WRITE(6,4079)
90*     4079 FORMAT('11')

```


1*	FUNCTION SATLUK(DIST)	000000
2*	ENTRY TASAN(A,B)	000000
3*	SATLUK=A+(B-A)*RANDOM(0)	000007
4*	RETURN	000012
5*	ENTRY EXPJA1(EX)	000012
6*	SATLUK==EX* LOG(RANDOM(0))	000023
7*	RETURN	000026
8*	ENTRY ORJAK(EX,STDX)	000026
9*	SUM=0.	000031
10*	DO 150 I=1,12	000031
11*	150 SUM=SUM+RANDOM(0)	000036
12*	SATLUK=STDX*(SUM=6.)*EX	000042
13*	RETURN	000045
14*	ENTRY POISSN(P)	000045
15*	X=0.	000046
16*	C=1./EXP(P)	000054
17*	TR=1,M	000057
18*	162 R1=RANDOM(0)	000062
19*	TR=TR*R1	000064
20*	IF (TR=C) 160,161,161	000067
21*	161 X=X+1.	000072
22*	GO TO 162	000074
23*	160 SATLUK=X	000075
24*	RETURN	000100
25*	ENTRY EXPJA2(N,PAR)	000100
26*	DIMENSION PAR(N),EX1(2), THIN(2),TN(2)	000100
27*	M=PAR(1)	000112
28*	DO 170 I=1,M	000134
29*	EX1(I)=PAR(I+1)	000135
30*	THIN(I)=PAR(I+M+1)	000137
31*	170 TN(I)=PAR(I+2*M+1)	000142
32*	R1=RANDOM(0)	000146
33*	TN1=0	000152
34*	DO 171 I=1,M	000152
35*	TN1=TN1+TN(I)	000155
36*	IF (R1.LE.TN1) GO TO 172	000163
37*	171 CONTINUE	000163
38*	WRITE(6,199) I	000172
39*	199 FORMAT(1 VIRHETILANNE; EXPJA2;N GENEROINNISSA (FRMT 199)1,14)	000172
40*	172 SATLUK=TN1(I)-(EX1(I)-TN1(I))*ALOG(1.-RANDOM(0))	000210
41*	RETURN	000277
42*	END	

1*	FUNCTION RANDOM(R)	000000
2*	DOUBLE PRECISION RND,RDF	000000
3*	RDF=4147555700	000000
4*	RND=RND/1638400+100/1638400/1638400	000001
5*	RND=DMOD(RND*RDF,1)	000005
6*	RANDOM=SNGL(RND)	000015
7*	RETURN	000017
8*	END	000030

7.4

Esimerkkiajon tulostuslistaus

koodimerkinnot:

sisääntulokaistat 01...13

ulosmenokaistat 14...25

konfliktipisteet K01...K72

generointipisteet G01...G13

poistumispisteet P14...P25

Kuva 7. - 1. Esimerkkiajon risteyksen kooditus

RISTEYKSEN DIMENSIOT

SISAANTULOKAISTOJEN LUKUMAARA:	13
ULOSHENOKAISTOJEN LUKUMAARA:	12
AJOURIEN LUKUMAARA:	42
AJOREITTIIEN LUKUMAARA:	25
KONFLIKTIPISTEIDEN LUKUMAARA:	72
LIIKENNEVALOSUUNTIEN LUKUMAARA:	21

REITTIKOHTAISET SYÖTTÖTIEDOT

REITTI	AJOURAT	REITTIÄ KULKEVAN LIIKENNEMAARAN OSUUS GENER-PISTEESSÄ GENEROIDUSTA MAARASTA
1	1 26 23 0 0 0	.90
2	1 25 0 0 0 0	.10
3	2 22 0 0 0 0	1.00
4	3 27 28 19 0 0	1.00
5	4 32 21 0 0 0	.40
6	4 33 18 0 0 0	.60
7	5 17 0 0 0 0	.70
8	5 40 38 25 0 0	.30
9	6 39 24 0 0 0	.50
10	6 39 22 0 0 0	.50
11	7 36 14 0 0 0	.55
12	7 37 21 0 0 0	.45
13	8 20 0 0 0 0	.80
14	8 41 17 0 0 0	.20
15	9 16 0 0 0 0	1.00
16	10 30 39 22 0 0	.15
17	10 30 39 24 0 0	.15
18	10 30 31 38 25 0	.15
19	10 30 31 15 0 0	.55
20	11 34 18 0 0 0	.45
21	11 35 14 0 0 0	.55
22	12 42 20 0 0 0	.60
23	12 23 0 0 0 0	.40
24	13 27 28 19 0 0	.70
25	13 27 30 31 15 0	.30

LIIKENNEVALO-OHJAUKSEN SYÖTTÖTIEDOT

LIIKENNEVALOJEN TILA SISAANTULOKAISTOJTTAIN SIMULOINNIN ALUSSA JA VAIHEKAAVION KOODITUS

VIHREA =1
 VIHREA-KELTAINEN =2
 PUNAINEN =3
 PUNAINEN-KELTAINEN =4

URA	TILA	KESTANYT	VAIHEKAAVIO VIH V-KEL PUN P-KEL
1	3	68	45 3 40 2
2	3	68	45 3 40 2
3	1	4	10 3 75 2
4	3	68	37 3 48 2
5	3	10	23 3 62 2
6	3	10	23 3 62 2
7	3	68	12 3 73 2
8	3	102	20 3 65 2
9	3	102	20 3 65 2
10	3	102	20 3 65 2
11	3	68	12 3 73 2
12	3	6	25 3 60 2
13	3	6	25 3 60 2
14	3	20	25 0 65 0
15	1	90	53 0 37 0
16	3	92	35 0 55 0
17	3	116	16 0 74 0
18	1	58	32 0 58 0
19	3	6	30 0 60 0
20	1	0	52 0 38 0
21	1	0	52 0 38 0

AJOURAKONTAISET SYÖTTÖTIEDOT

AJOURA	SEURAJA- URAT		EDELTAJX- URAT		URAN PITUUS	MAX.NOPEUS URALLA 15	GENEROINNIN AIKAVÄLIT 2-HUIPPU-EXP.-JAK.				AJONEUVOLAJI 1 OSUUS PITUUS		AJONEUVOLAJI 2 OSUUS PITUUS		AJONEUVOLAJI 3 OSUUS PITUUS		
	SUOR KAA	SUOR KAA	SUOR KAA	SUOR KAA			KAI	KAZ	MIN1	MIN2	OS.	KA	HAY	KA	HAY	KA	HAY
1	0	0	0	26	53.0	10.0	9.	4.	8.	3.	7.	1.	1.5	5.	5.0	5.	10.0
2	0	0	22	25	53.0	10.0	12.	6.	10.	6.	4.	1.0	6.0	0.	0.	0.	0.
3	0	0	30	27	60.0	4.0	10.	5.	7.	5.	6.	0.7	6.0	0.	0.	0.	0.
4	0	0	32	33	54.0	7.0	20.	2.	17.	1.	8.	0.4	12.0	0.	0.	0.	0.
5	0	0	40	17	50.0	9.0	5.	12.	3.	10.	0.	1.	1.5	6.	5.5	0.	10.0
6	0	0	39	0	56.0	9.0	8.	5.	5.	4.	1.0	0.8	5.5	5.	1.	0.	0.
7	0	0	37	36	54.0	7.0	7.	7.	3.	3.	0.	0.5	12.0	0.	0.	0.	0.
8	0	0	41	20	50.0	10.0	8.	2.	6.	1.	5.	1.0	5.5	0.	0.	0.	0.
9	0	0	16	0	53.0	10.0	11.	3.	7.	1.	0.	1.0	5.5	0.	0.	0.	0.
10	0	0	30	0	53.0	10.0	10.	30.	0.	0.	0.	0.5	10.0	0.	0.	0.	0.
11	0	0	35	34	52.0	8.0	14.	5.	8.	0.	0.	0.6	12.0	0.	0.	0.	0.
12	0	0	42	23	50.0	10.0	19.	9.	8.	4.	0.	1.0	5.5	0.	0.	0.	0.
13	0	0	27	0	58.0	10.0	7.	7.	3.	3.	0.	0.5	12.0	0.	0.	0.	0.
14	35	36	0	0	15.0	9.0	9.0	0	0	0	0	0.5	12.0	0.	0.	0.	0.
15	31	0	0	0	15.0	10.0	10.0	0	0	0	0	0.5	12.0	0.	0.	0.	0.
16	9	28	0	0	15.0	10.0	10.0	0	0	0	0	0.5	12.0	0.	0.	0.	0.
17	41	5	0	0	15.0	10.0	10.0	0	0	0	0	0.5	12.0	0.	0.	0.	0.
18	33	34	0	0	15.0	9.0	9.0	0	0	0	0	0.5	12.0	0.	0.	0.	0.
19	28	0	0	0	15.0	10.0	10.0	0	0	0	0	0.5	12.0	0.	0.	0.	0.
20	42	8	0	0	15.0	10.0	10.0	0	0	0	0	0.5	12.0	0.	0.	0.	0.
21	32	37	0	0	15.0	9.0	9.0	0	0	0	0	0.5	12.0	0.	0.	0.	0.
22	2	39	0	0	15.0	10.0	10.0	0	0	0	0	0.5	12.0	0.	0.	0.	0.
23	26	12	0	0	15.0	10.0	10.0	0	0	0	0	0.5	12.0	0.	0.	0.	0.
24	39	0	0	0	15.0	6.0	4.0	0	0	0	0	0.5	12.0	0.	0.	0.	0.
25	38	1	0	0	15.0	9.0	9.0	0	0	0	0	0.5	12.0	0.	0.	0.	0.
26	1	0	0	0	11.0	9.0	9.0	0	0	0	0	0.5	12.0	0.	0.	0.	0.
27	13	3	0	0	10.0	7.0	7.0	0	0	0	0	0.5	12.0	0.	0.	0.	0.
28	27	0	0	0	3.0	8.0	8.0	0	0	0	0	0.5	12.0	0.	0.	0.	0.
29	0	0	0	0	10.0	10.0	10.0	0	0	0	0	0.5	12.0	0.	0.	0.	0.
30	10	27	0	0	9.0	10.0	10.0	0	0	0	0	0.5	12.0	0.	0.	0.	0.
31	30	0	15	38	3.0	10.0	10.0	0	0	0	0	0.5	12.0	0.	0.	0.	0.
32	4	0	21	0	9.0	5.0	9.0	0	0	0	0	0.5	12.0	0.	0.	0.	0.
33	0	0	18	0	9.0	3.0	9.0	0	0	0	0	0.5	12.0	0.	0.	0.	0.
34	0	11	0	18	6.0	3.0	3.0	0	0	0	0	0.5	12.0	0.	0.	0.	0.
35	11	0	14	0	10.0	5.0	5.0	0	0	0	0	0.5	12.0	0.	0.	0.	0.
36	0	7	0	14	5.0	2.0	2.0	0	0	0	0	0.5	12.0	0.	0.	0.	0.
37	7	0	0	21	12.0	4.0	4.0	0	0	0	0	0.5	12.0	0.	0.	0.	0.
38	40	31	25	0	18.0	8.0	8.0	0	0	0	0	0.5	12.0	0.	0.	0.	0.
39	6	30	24	22	13.0	8.0	8.0	0	0	0	0	0.5	12.0	0.	0.	0.	0.
40	5	0	38	0	6.0	8.0	8.0	0	0	0	0	0.5	12.0	0.	0.	0.	0.
41	8	0	17	0	15.0	11.0	11.0	0	0	0	0	0.5	12.0	0.	0.	0.	0.
42	12	0	20	0	22.0	9.0	9.0	0	0	0	0	0.5	12.0	0.	0.	0.	0.

KONFLIKTIPISTEKOHTAISET SYÖTTÖTIEDOT

KONFLIKTI- PISTE	LEIKKAAVAT AJOURAT	K-PISTEEN ETÄISYYS EJELL.URAN ALUSTA	JALK.URAN ALUSTA	URIEN KONTAUSKULMAT JALK.URAN TULOSEKTORI EJELL.URAN SUHTEEN	EJELL.URAN TULOSEKTORI JALK.URAN SUHTEEN	NAKOESTEEN ETÄISYYS KONFLIKTIPISTEESTÄ MITATTUNA URAA PITKIN
1	1 - 38	100	15	5	2	15
2	24 - 26	3	3	1	6	15
3	13 - 26	50	7	5	2	15
4	12 - 26	50	10	5	2	15
5	2 - 38	50	13	5	2	15
6	2 - 39	53	13	5	2	15
7	13 - 22	53	7	5	2	15
8	22 - 42	10	3	5	2	15
9	3 - 38	50	10	5	2	15
10	3 - 39	53	10	5	2	15
11	3 - 13	60	58	1	6	15
12	4 - 38	50	7	5	2	15
13	4 - 39	53	7	5	2	15
14	27 - 32	3	6	5	2	15
15	32 - 37	7	12	6	1	15
16	21 - 42	1	9	1	6	15
17	5 - 41	50	15	5	2	15
18	16 - 40	12	3	1	6	15
19	31 - 40	3	6	1	6	15
20	14 - 38	4	4	1	6	15
21	14 - 39	1	4	1	6	15
22	6 - 30	56	9	5	2	15
23	6 - 16	53	9	5	2	15
24	6 - 41	50	12	5	2	15
25	7 - 41	50	9	5	2	15
26	7 - 16	53	6	5	2	15
27	8 - 42	50	22	5	2	15
28	19 - 41	3	3	1	6	15
29	18 - 41	4	6	1	6	15
30	9 - 42	50	19	5	2	15
31	10 - 42	50	16	5	2	15
32	11 - 42	50	12	5	2	15
33	9 - 28	53	3	5	2	15
34	10 - 27	53	10	5	2	15
35	16 - 18	3	1	5	2	15
36	33 - 34	9	6	6	1	15
37	30 - 33	3	8	5	2	15
38	30 - 34	2	5	6	1	15
39	30 - 37	5	1	2	5	15
40	30 - 36	6	1	1	6	15
41	33 - 37	5	6	2	5	15
42	35 - 36	10	5	6	1	15
43	27 - 34	9	1	1	6	15
44	33 - 35	4	6	2	5	15
45	35 - 37	4	7	2	5	15
46	27 - 35	7	1	1	6	15
47	27 - 37	4	11	4	3	15
48	1 - 43	47	1	3	4	15
49	2 - 43	47	2	3	4	15
50	3 - 43	47	3	3	4	15
51	4 - 43	47	4	3	4	15
52	17 - 44	3	1	3	4	15
53	16 - 44	15	2	3	4	15
54	15 - 44	3	3	3	4	15
55	14 - 44	7	4	3	4	15
56	5 - 45	47	1	3	4	15
57	6 - 45	47	2	3	4	15
58	7 - 45	47	3	3	4	15
59	20 - 46	3	1	3	4	15
60	19 - 46	6	2	3	4	15
61	18 - 46	7	3	3	4	15
62	8 - 47	47	1	3	4	15
63	9 - 47	47	2	3	4	15
64	10 - 47	47	3	3	4	15
65	11 - 47	47	4	3	4	15
66	21 - 48	4	3	3	4	15
67	22 - 48	12	2	3	4	15
68	23 - 48	3	1	3	4	15
69	12 - 49	47	1	3	4	15
70	13 - 49	47	2	3	4	15
71	24 - 50	6	2	3	4	15
72	25 - 50	3	1	3	4	15

AJOURAKOHTAISET KONFLIKTIPISTELUETTELOT

AJOURA KONFLIKTIPISTEET

1	48	1	0	0	0
2	49	5	4	0	0
3	50	9	10	11	0
4	51	12	13	0	0
5	56	17	0	0	0
6	57	24	23	22	0
7	58	25	26	0	0
8	62	27	0	0	0
9	63	30	33	0	0
10	64	31	34	0	0
11	65	32	0	0	0
12	69	4	0	0	0
13	70	3	7	11	0
14	21	20	55	0	0
15	54	0	0	0	0
16	35	26	23	18	53
17	52	0	0	0	0
18	35	29	41	0	0
19	28	40	0	0	0
20	59	0	0	0	0
21	16	66	0	0	0
22	7	8	67	0	0
23	48	0	0	0	0
24	2	71	0	0	0
25	72	0	0	0	0
26	2	3	4	0	0
27	14	47	46	43	34
28	33	0	0	0	0
29	0	0	0	0	0
30	38	37	39	40	22
31	19	0	0	0	0
32	14	15	0	0	0
33	44	41	37	36	0
34	43	38	36	0	0
35	46	45	44	42	0
36	40	42	0	0	0
37	39	41	45	47	15
38	20	12	9	5	1
39	21	13	10	4	0
40	18	19	0	0	0
41	28	29	25	24	17
42	8	16	32	31	73
43	30	27	0	0	0

JATKOLUETTELO

TULOSTETAAN KUMULATIIVISIA TILASTOJA, KUH SIMULOINTI ON KESTANYT 300. SEKUNTIA

KONFLIKTI- PISTL	KONFLIKTIEN LUKUMAARA	KESKIMAAR. TAPAHTUMIS- AIKAVALI	KESKIMAAR. NOPEUS ENHEN KONFLIKTIA	JAKAUTUMIN JAKAUTUMAT	<-9.0	<-8.0	<-7.0	<-6.0	<-5.0	<-4.0	<-3.0	<-2.0	<-1.0	<-0.0
1	0	.0	.0	0	0	0	0	0	0	0	0	0	0	0
2	7	40.9	3.4	2	0	0	0	0	1	1	2	0	1	1
3	10	28.3	2.7	1	0	0	0	2	1	0	1	2	3	3
4	7	28.9	4.5	1	0	0	0	0	0	0	0	2	4	4
5	0	.0	.0	0	0	0	0	0	0	0	0	0	0	0
6	14	19.0	3.2	1	0	0	0	0	0	1	1	3	7	7
7	6	40.2	2.8	0	0	1	0	0	0	0	2	1	2	2
8	5	52.7	1.0	1	0	0	0	0	0	0	0	0	1	1
9	2	87.5	4.2	0	0	0	0	0	0	0	0	1	1	1
10	4	70.0	2.3	0	0	0	0	0	0	0	0	1	3	3
11	6	47.8	4.4	0	0	0	0	0	0	0	1	3	2	2
12	4	31.1	2.9	0	0	0	0	0	1	0	0	1	2	2
13	6	36.9	2.0	2	0	0	0	0	0	1	0	0	2	2
14	0	.0	.0	0	0	0	0	0	0	0	0	0	0	0
15	2	69.0	5.5	0	0	0	0	0	0	1	1	0	0	0
16	2	122.5	3.3	0	0	0	0	0	0	0	0	1	1	1
17	4	60.4	4.1	1	0	0	0	0	0	1	0	2	0	0
18	0	.0	.0	0	0	0	0	0	0	0	0	0	0	0
19	2	80.5	4.2	1	1	0	0	0	0	0	0	0	0	0
20	0	.0	.0	0	0	0	0	0	0	0	0	0	0	0
21	0	.0	.0	0	0	0	0	0	0	0	0	0	0	0
22	4	38.7	5.4	2	0	0	0	0	0	1	0	1	0	0
23	2	130.5	4.3	1	0	1	0	0	0	0	0	0	0	0
24	0	.0	.0	0	0	0	0	0	0	0	0	0	0	0
25	0	.0	.0	0	0	0	0	0	0	0	0	0	0	0
26	2	115.7	3.2	1	0	0	0	1	0	0	0	0	0	0
27	9	33.2	3.3	5	3	0	0	0	0	0	0	1	0	0
28	1	108.5	4.1	1	0	0	0	0	0	0	0	0	0	0
29	2	21.7	2.6	0	0	0	0	0	0	0	0	0	2	2
30	4	43.1	3.0	0	0	0	0	0	0	0	1	0	3	3
31	2	34.2	4.8	0	0	0	0	0	0	0	0	1	1	1
32	2	123.5	2.9	1	0	0	0	0	0	0	0	1	0	0
33	7	42.8	2.2	3	0	0	0	0	0	0	0	0	2	2
34	6	49.2	3.6	0	1	0	0	0	0	1	1	0	1	1
35	4	59.0	2.7	2	0	0	0	0	0	0	0	0	1	1
36	4	58.7	4.6	2	0	0	0	0	0	1	1	0	0	0
37	1	199.0	3.8	0	0	0	0	0	0	1	0	0	0	0
38	0	.0	.0	0	0	0	0	0	0	0	0	0	0	0
39	0	.0	.0	0	0	0	0	0	0	0	0	0	0	0
40	0	.0	.0	0	0	0	0	0	0	0	0	0	0	0
41	0	.0	.0	0	0	0	0	0	0	0	0	0	0	0
42	1	133.5	5.1	0	0	0	0	0	0	0	0	1	0	0
43	0	.0	.0	0	0	0	0	0	0	0	0	0	0	0
44	2	72.7	2.0	0	0	0	0	0	0	0	1	1	0	0
45	2	70.2	5.0	1	0	0	0	1	0	0	0	0	0	0
46	0	.0	.0	0	0	0	0	0	0	0	0	0	0	0
47	0	.0	.0	0	0	0	0	0	0	0	0	0	0	0
48	0	.0	.0	0	0	0	0	0	0	0	0	0	0	0
49	0	.0	.0	0	0	0	0	0	0	0	0	0	0	0
50	0	.0	.0	0	0	0	0	0	0	0	0	0	0	0
51	0	.0	.0	0	0	0	0	0	0	0	0	0	0	0
52	9	28.0	1.9	2	1	0	1	1	0	0	0	2	3	3
53	5	47.6	4.5	1	0	0	1	0	0	0	0	0	0	0
54	4	50.1	4.3	3	0	0	0	0	0	0	0	1	0	0
55	1	135.0	4.3	0	0	0	0	0	1	0	0	0	0	0
56	0	.0	.0	0	0	0	0	0	0	0	0	0	0	0
57	0	.0	.0	0	0	0	0	0	0	0	0	0	0	0
58	0	.0	.0	0	0	0	0	0	0	0	0	0	0	0
59	7	41.3	1.7	0	0	0	0	0	0	0	2	2	2	2
60	7	39.4	3.9	1	0	0	0	1	0	1	0	2	2	2
61	14	21.4	2.1	1	0	0	0	0	0	0	0	5	6	6
62	0	.0	.0	0	0	0	0	0	0	0	0	0	0	0
63	0	.0	.0	0	0	0	0	0	0	0	0	0	0	0
64	0	.0	.0	0	0	0	0	0	0	0	0	0	0	0
65	0	.0	.0	0	0	0	0	0	0	0	0	0	0	0
66	3	96.7	5.7	2	1	0	0	0	0	0	0	0	0	0
67	22	12.6	2.6	0	0	0	0	0	0	0	0	1	17	17
68	5	32.5	2.8	2	0	0	0	1	0	0	0	0	2	2
69	0	.0	.0	0	0	0	0	0	0	0	0	0	0	0
70	0	.0	.0	0	0	0	0	0	0	0	0	0	0	0
71	2	72.0	4.6	1	0	0	0	0	0	0	1	0	0	0
72	2	91.0	5.9	2	0	0	0	0	0	0	0	0	0	0
YHTEENSA	217			44	7	3	7	4	10	15	36	74		

AJONEUVULUOKKIEN VALINEN KONFLIKTIJAKAUTUMA
KOKO RISTEYKSESSÄ, HIDASTUVUUKSIEN OLLESSA ALLE -5.0

KONFL. AJON. LUOKKA		1	2	3	4	5	6
	
K	1	15	4	4	3	0	14
A	2	3	0	2	1	0	1
S	3	3	0	2	1	0	5
A	4	3	0	1	1	0	1
J	5	0	0	0	0	0	1
O	6	0	0	0	0	0	0
N							
L							
K	6	0	0	0	0	0	0

AJONEUVULUOKKIEN VALINEN KONFLIKTIJAKAUTUMA
KOKO RISTEYKSESSÄ, HIDASTUVUUKSIEN OLLESSA ALLE -2.0

KONFL. AJON. LUOKKA		1	2	3	4	5	6
	
K	1	29	9	7	7	0	25
A	2	5	0	4	3	0	2
S	3	6	0	4	1	0	8
A	4	4	0	4	3	0	3
J	5	0	1	0	0	0	1
O	6	0	0	0	0	0	0
N							
L							
K	6	0	0	0	0	0	0

AJONEUVULUOKKIEN VALINEN KONFLIKTIJAKAUTUMA
KONFLIKTIPISTEESSÄ 7 HIDASTUVUUKSIEN OLLESSA ALLE -5.0

KONFL. AJON. LUOKKA		1	2	3	4	5	6
	
K	1	0	0	0	1	0	0
A	2	0	0	0	0	0	0
S	3	0	0	0	0	0	0
A	4	0	0	0	0	0	0
J	5	0	0	0	0	0	0
O	6	0	0	0	0	0	0
N							
L							
K	6	0	0	0	0	0	0

AJONEUVULUOKKIEN VALINEN KONFLIKTIJAKAUTUMA
KONFLIKTIPISTEESSÄ 7 HIDASTUVUUKSIEN OLLESSA ALLE -2.0

KONFL. AJON. LUOKKA		1	2	3	4	5	6
	
K	1	0	0	1	2	0	0
A	2	0	0	0	0	0	0
S	3	1	0	0	0	0	0
A	4	0	0	0	0	0	0
J	5	0	0	0	0	0	0
O	6	0	0	0	0	0	0
N							
L							
K	6	0	0	0	0	0	0

AJONEUVJLUOKKIEN VALINEN KONFLIKTIJAKAUTUMA
 KONFLIKTIPISTEESSA 27 HIDASTUVUUKSIEN OLLESSA ALLE -5.0

		KONFL. AJON. LUOKKA					
		1	2	3	4	5	6
	
K	1	8	0	0	0	0	0
A	2	0	0	0	0	0	0
S	3	0	0	0	0	0	0
A	4	0	0	0	0	0	0
J	5	0	0	0	0	0	0
O	6	0	0	0	0	0	0
N							
L							
K							

AJONEUVJLUOKKIEN VALINEN KONFLIKTIJAKAUTUMA
 KONFLIKTIPISTEESSA 27 HIDASTUVUUKSIEN OLLESSA ALLE -2.0

		KONFL. AJON. LUOKKA					
		1	2	3	4	5	6
	
K	1	9	0	0	0	0	0
A	2	0	0	0	0	0	0
S	3	0	0	0	0	0	0
A	4	0	0	0	0	0	0
J	5	0	0	0	0	0	0
O	6	0	0	0	0	0	0
N							
L							
K							

SISÄÄNTULOKAISTOILLA KESKIMÄÄRISET MAKSIMIJONONPITUUDET

1	2	3	4	5	6	7	8	9	10	11	12	13
2	1	5	1	2	3	2	2	4	2	2	1	3